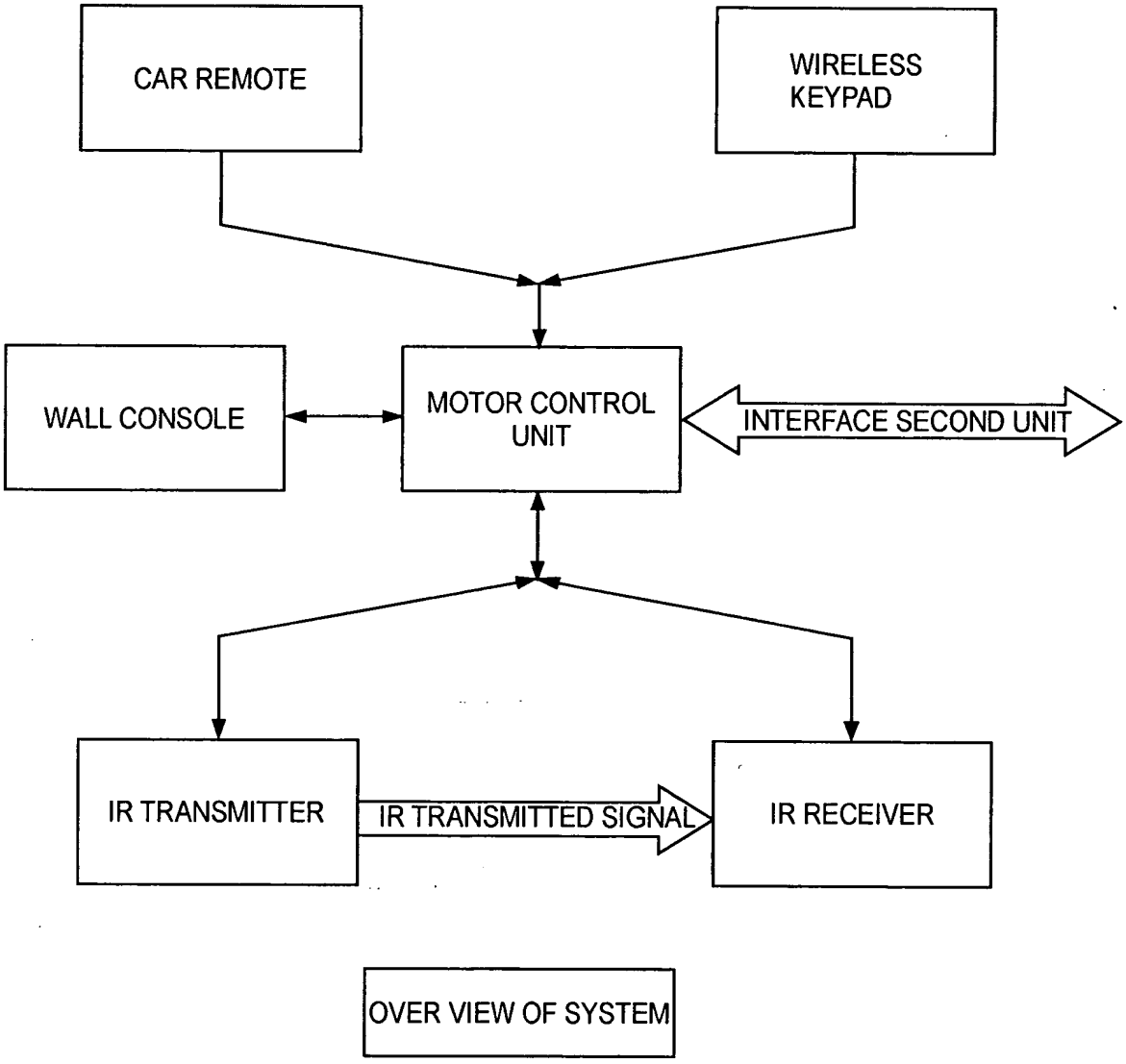
OIPE
MAY 23 2005
JC112 3005
PATENT & TRADEMARK OFFICE

Fig. 1



OVER VIEW OF SYSTEM

Fig. 2

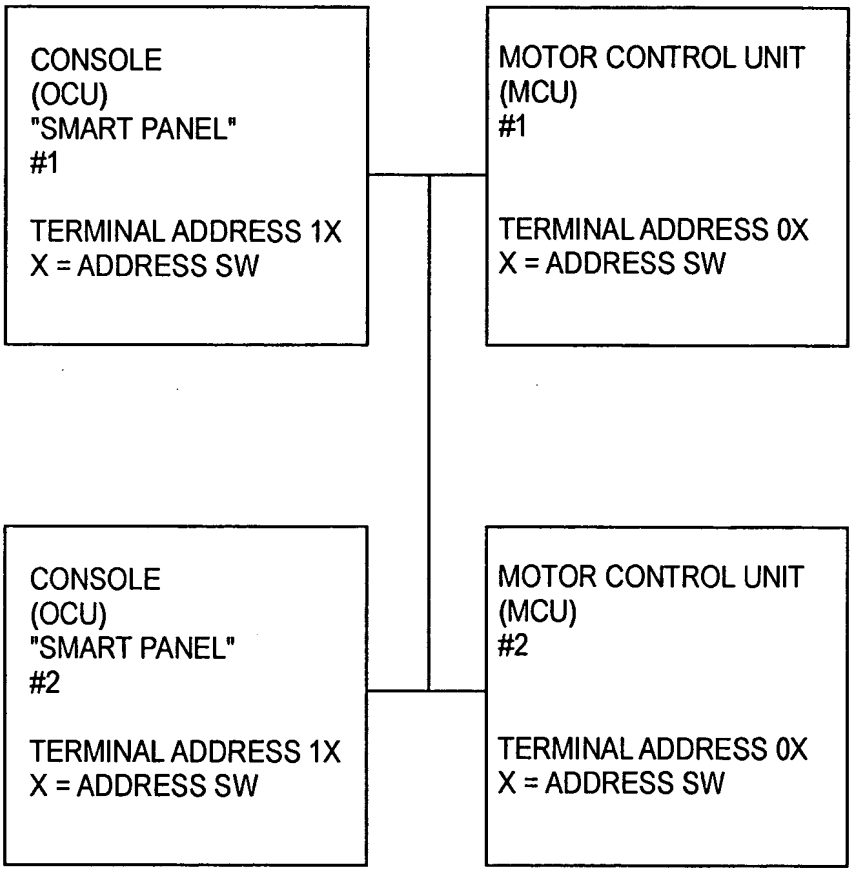

CONSOLE
(OCU)
"SMART PANEL"
#1
TERMINAL ADDRESS 1X
X = ADDRESS SW
MOTOR CONTROL UNIT
(MCU)
#1
TERMINAL ADDRESS 0X
X = ADDRESS SW
CONSOLE
(OCU)
"SMART PANEL"
#2
TERMINAL ADDRESS 1X
X = ADDRESS SW
MOTOR CONTROL UNIT
(MCU)
#2
TERMINAL ADDRESS 0X
X = ADDRESS SW

Fig. 3

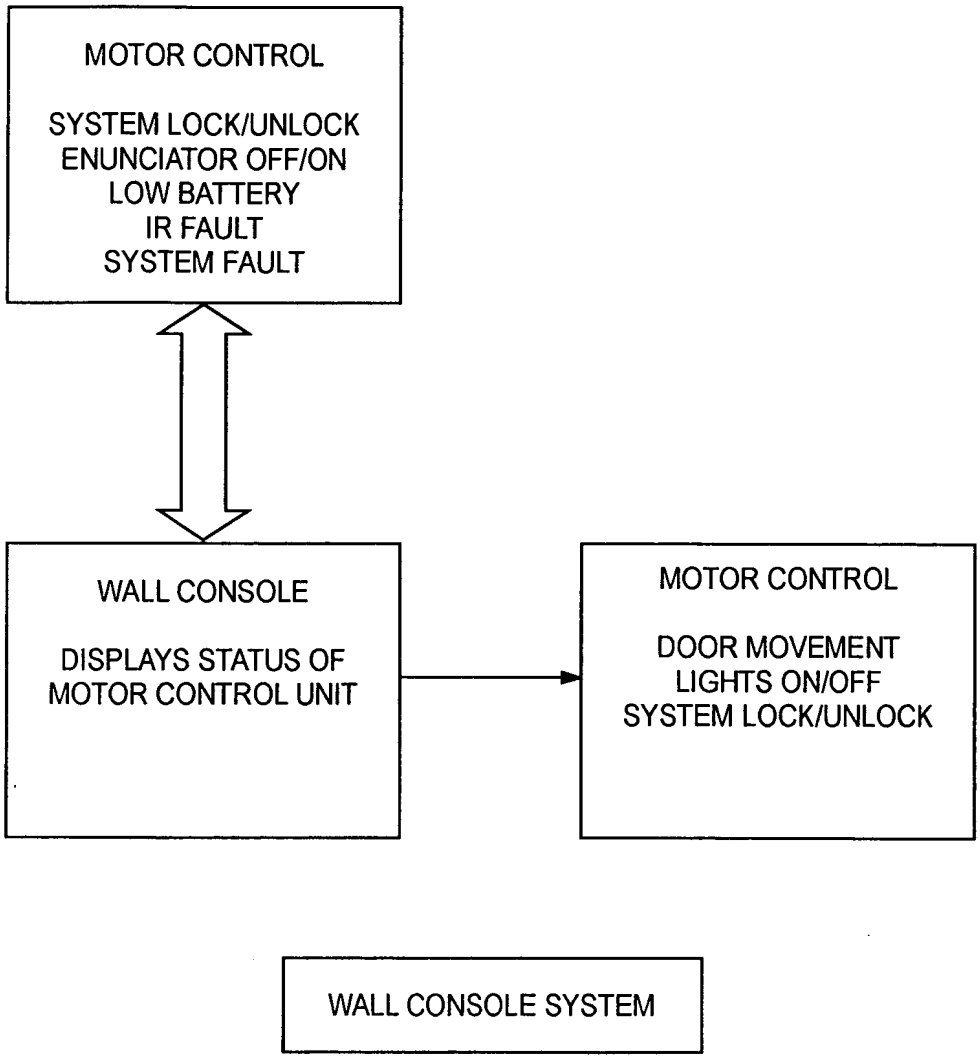

MOTOR CONTROL
SYSTEM LOCK/UNLOCK
ENUNCIATOR OFF/ON
LOW BATTERY
IR FAULT
SYSTEM FAULT
WALL CONSOLE
DISPLAYS STATUS OF
MOTOR CONTROL UNIT
MOTOR CONTROL
DOOR MOVEMENT
LIGHTS ON/OFF
SYSTEM LOCK/UNLOCK
WALL CONSOLE SYSTEM

Fig. 4

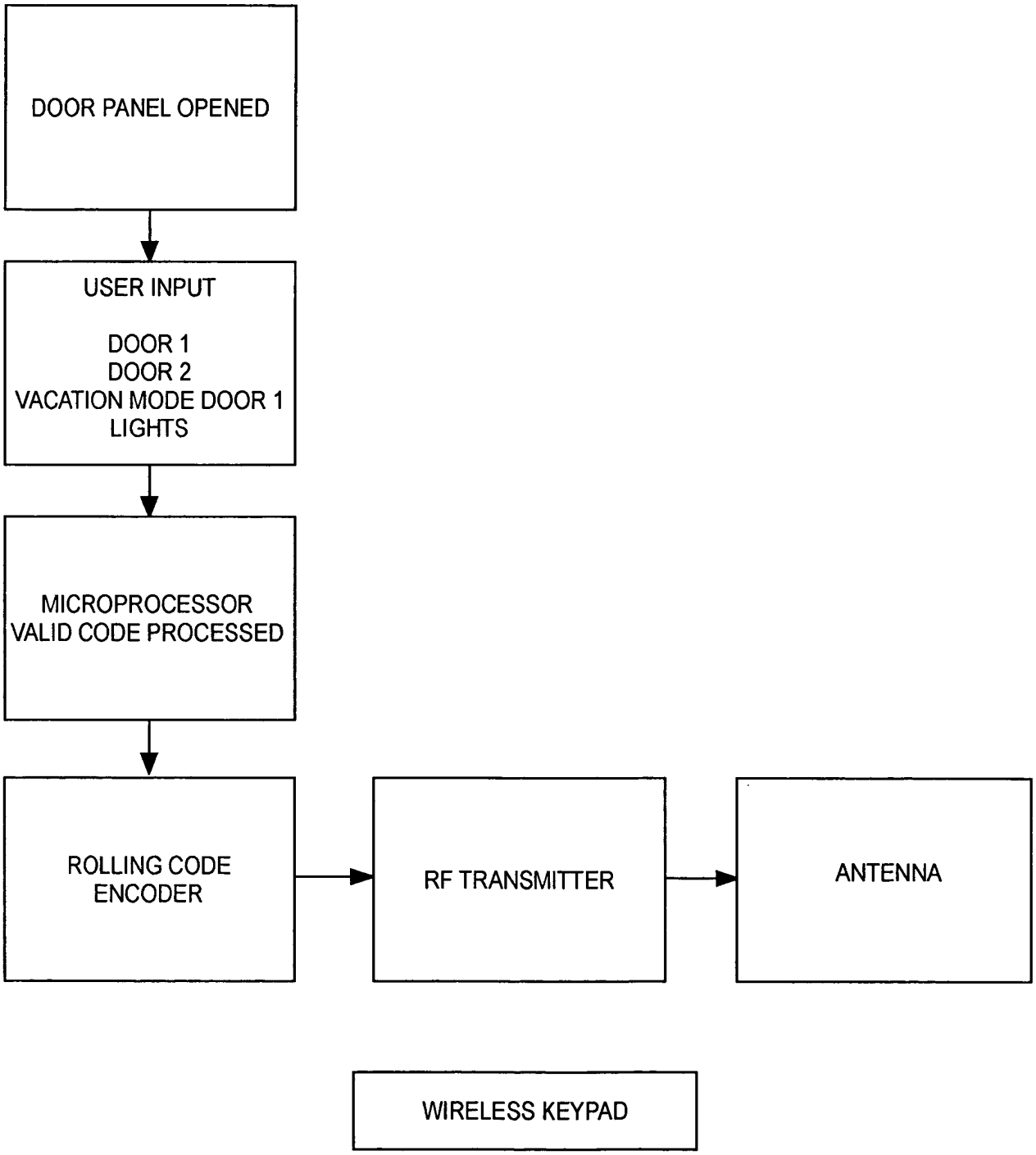

DOOR PANEL OPENED
USER INPUT
DOOR 1
DOOR 2
VACATION MODE DOOR 1
LIGHTS
MICROPROCESSOR
VALID CODE PROCESSED
ROLLING CODE
ENCODER
RF TRANSMITTER
ANTENNA
WIRELESS KEYPAD

Fig. 5

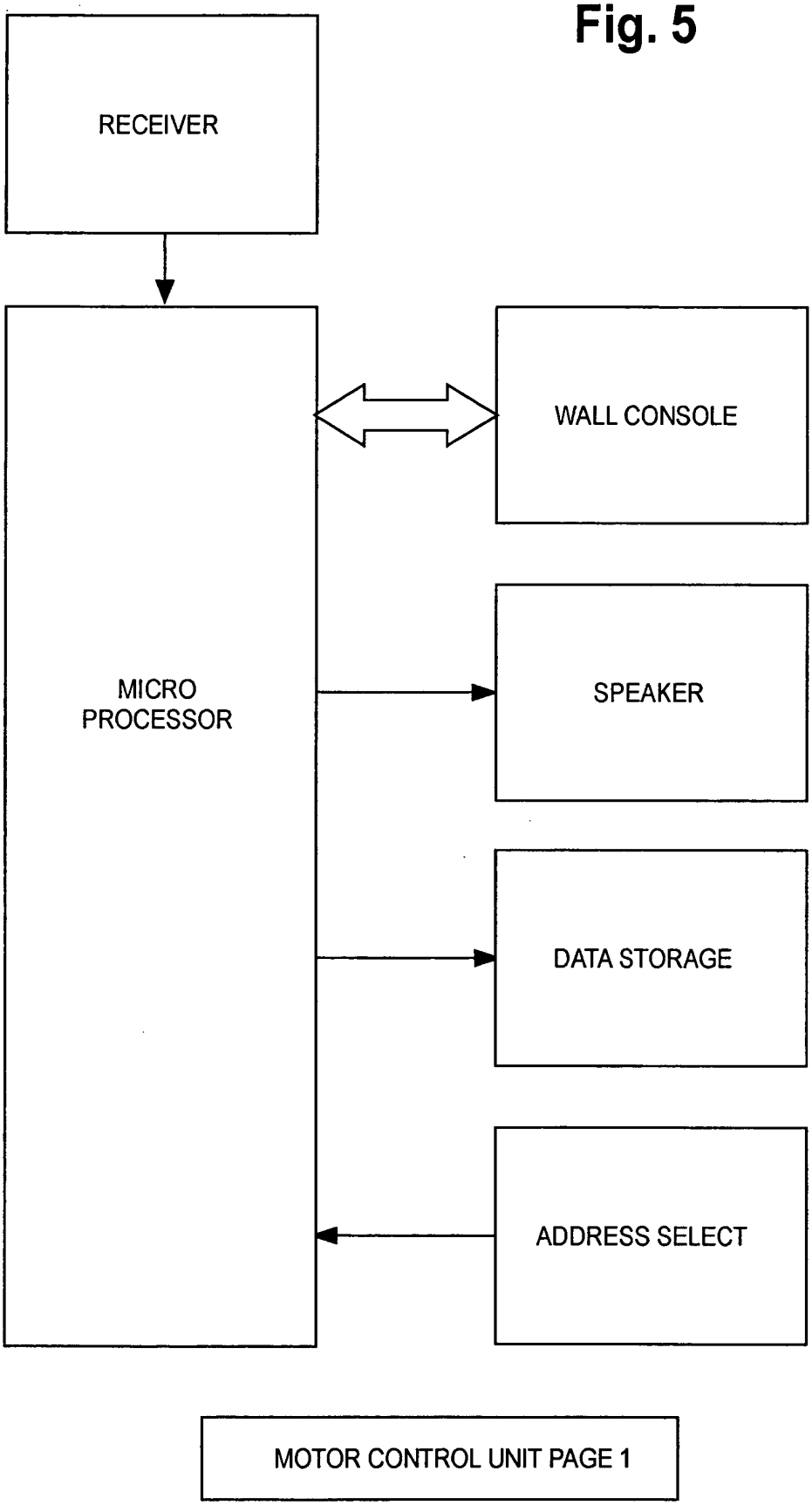
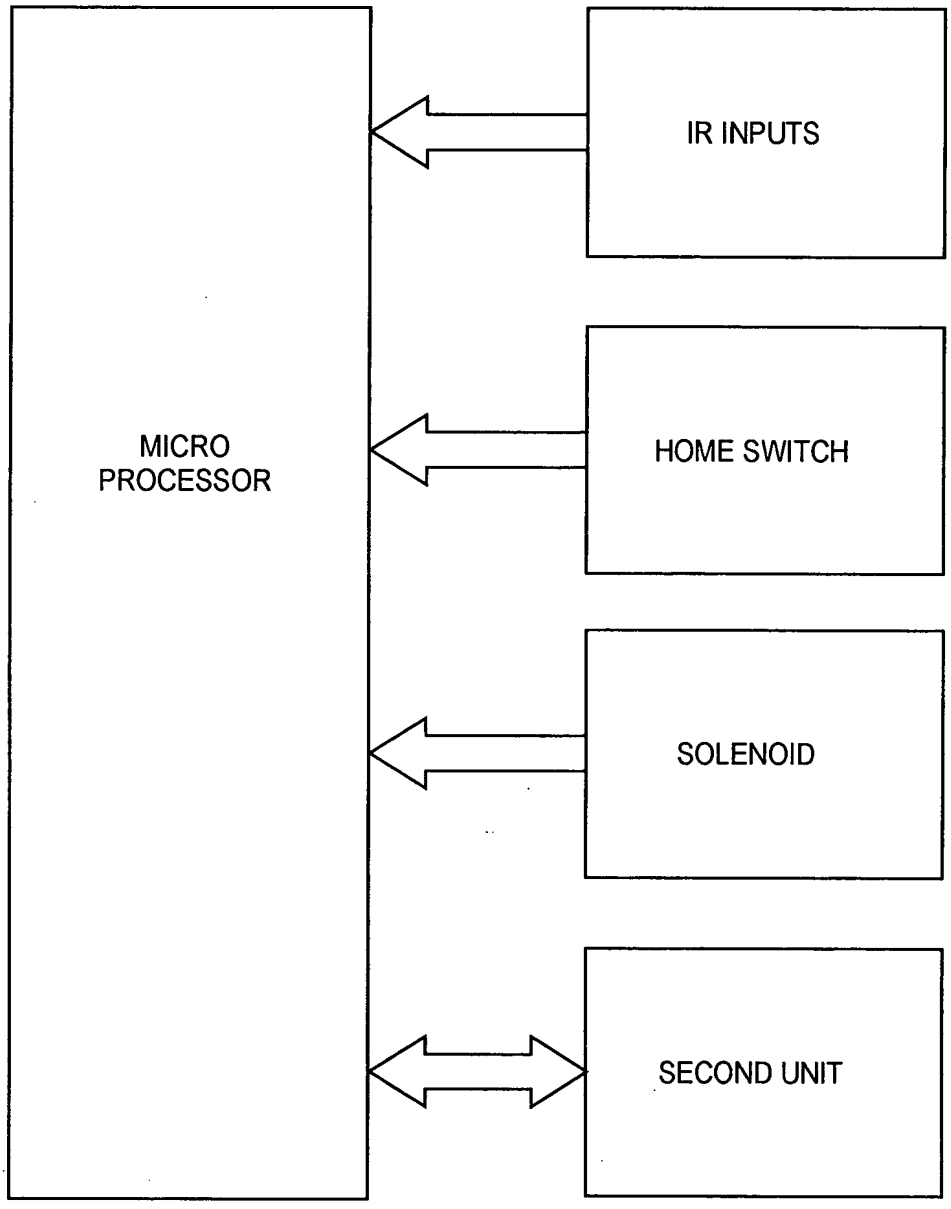

RECEIVER
WALL CONSOLE
MICRO
PROCESSOR
SPEAKER
DATA STORAGE
ADDRESS SELECT
MOTOR CONTROL UNIT PAGE 1

Fig. 6



MOTOR CONTROL UNIT PAGE 2

Fig. 7

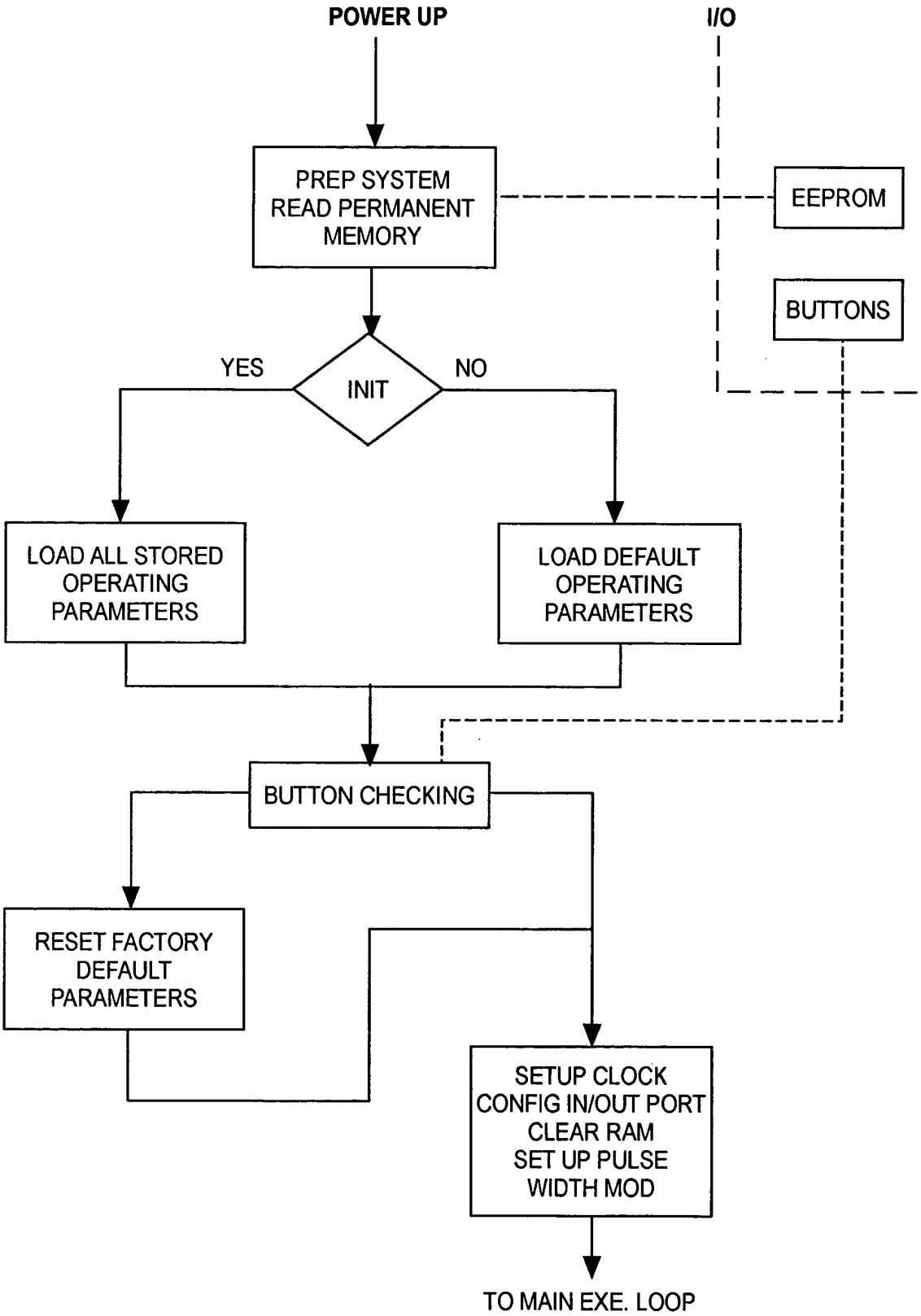

POWER UP
I/O
PREP SYSTEM
READ PERMANENT
MEMORY
EEPROM
BUTTONS
YES
INIT
NO
LOAD ALL STORED
OPERATING
PARAMETERS
LOAD DEFAULT
OPERATING
PARAMETERS
BUTTON CHECKING
RESET FACTORY
DEFAULT
PARAMETERS
SETUP CLOCK
CONFIG IN/OUT PORT
CLEAR RAM
SET UP PULSE
WIDTH MOD
TO MAIN EXE. LOOP

Fig. 8

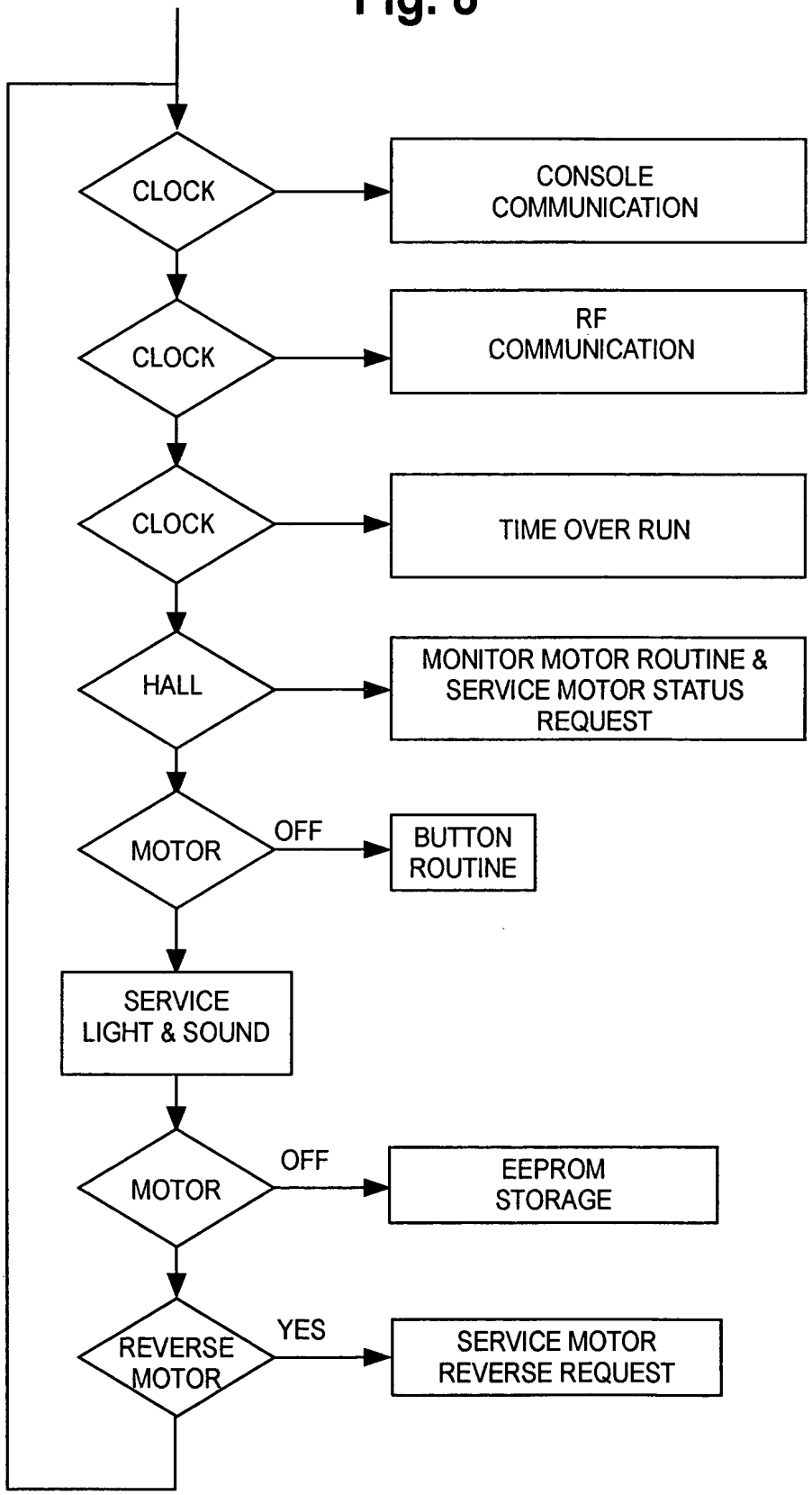

CLOCK
CONSOLE COMMUNICATION
CLOCK
RF COMMUNICATION
CLOCK
TIME OVER RUN
HALL
MONITOR MOTOR ROUTINE & SERVICE MOTOR STATUS REQUEST
MOTOR
OFF
BUTTON ROUTINE
SERVICE LIGHT & SOUND
MOTOR
OFF
EEPROM STORAGE
REVERSE MOTOR
YES
SERVICE MOTOR REVERSE REQUEST

Fig. 9

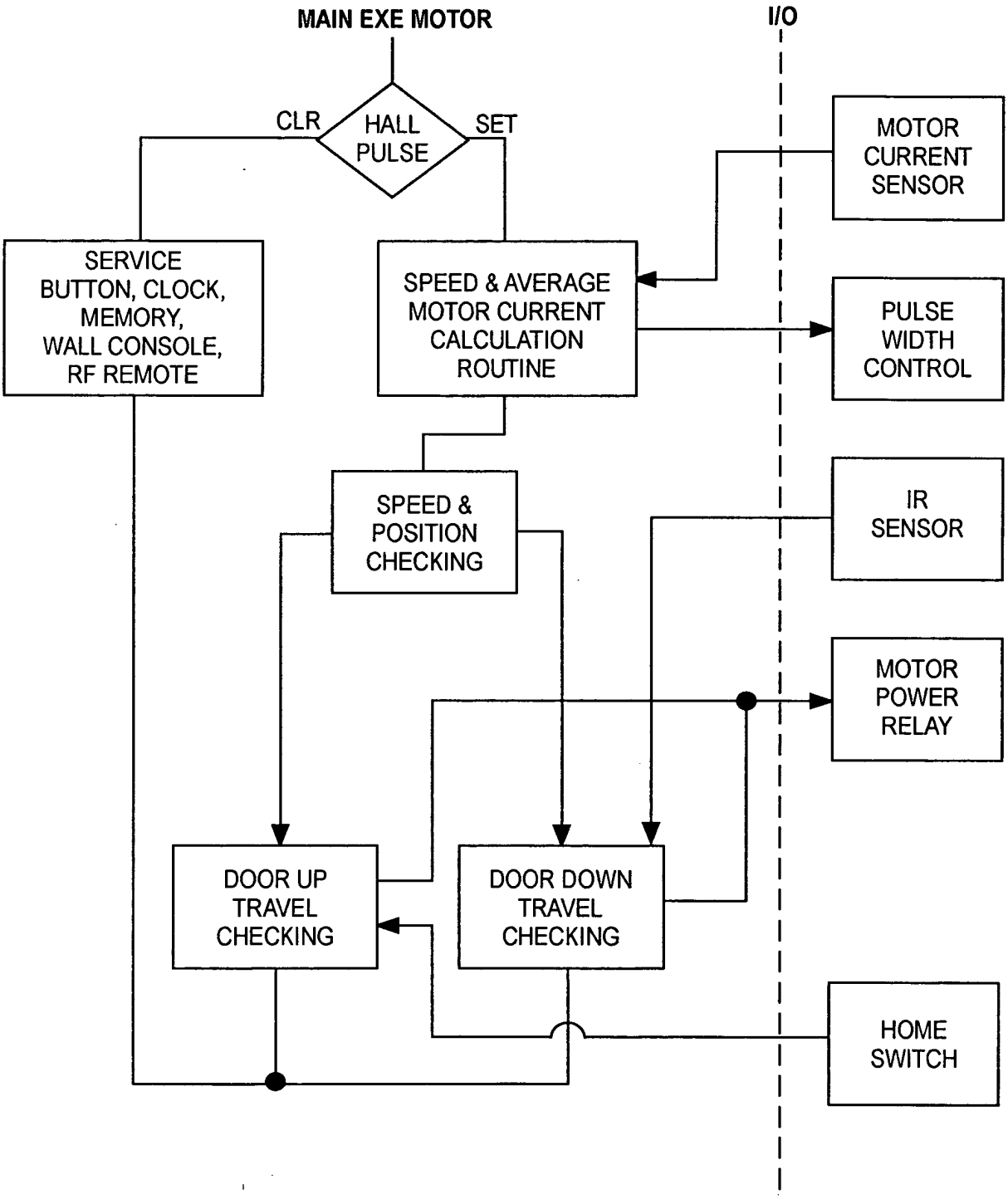

MAIN EXE MOTOR
I/O
CLR
HALL PULSE
SET
MOTOR CURRENT SENSOR
SERVICE BUTTON, CLOCK, MEMORY, WALL CONSOLE, RF REMOTE
SPEED & AVERAGE MOTOR CURRENT CALCULATION ROUTINE
PULSE WIDTH CONTROL
SPEED & POSITION CHECKING
IR SENSOR
MOTOR POWER RELAY
DOOR UP TRAVEL CHECKING
DOOR DOWN TRAVEL CHECKING
HOME SWITCH

Fig. 10

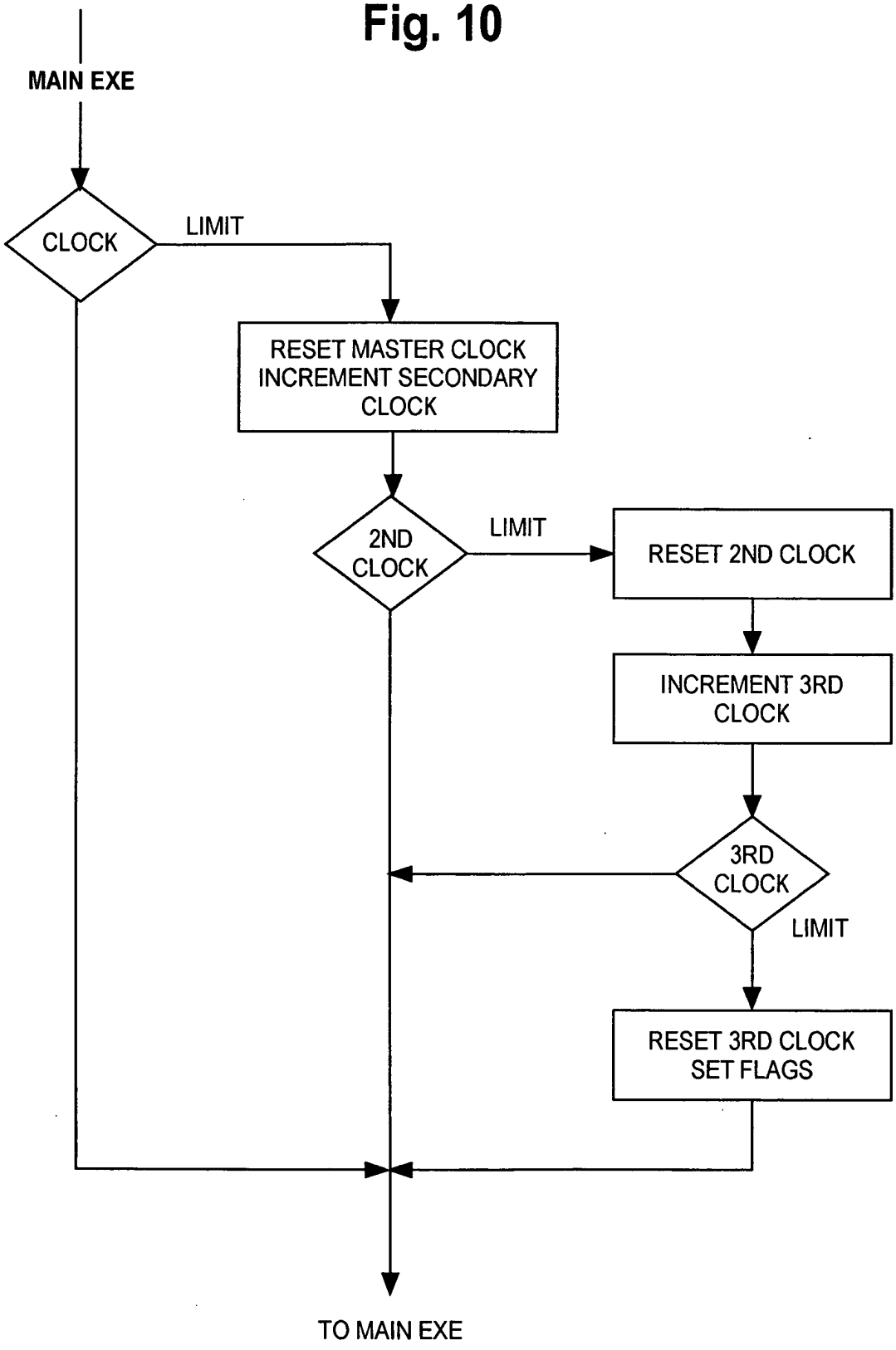

MAIN EXE
CLOCK
LIMIT
RESET MASTER CLOCK
INCREMENT SECONDARY
CLOCK
2ND
CLOCK
LIMIT
RESET 2ND CLOCK
INCREMENT 3RD
CLOCK
3RD
CLOCK
LIMIT
RESET 3RD CLOCK
SET FLAGS
TO MAIN EXE

Fig. 11

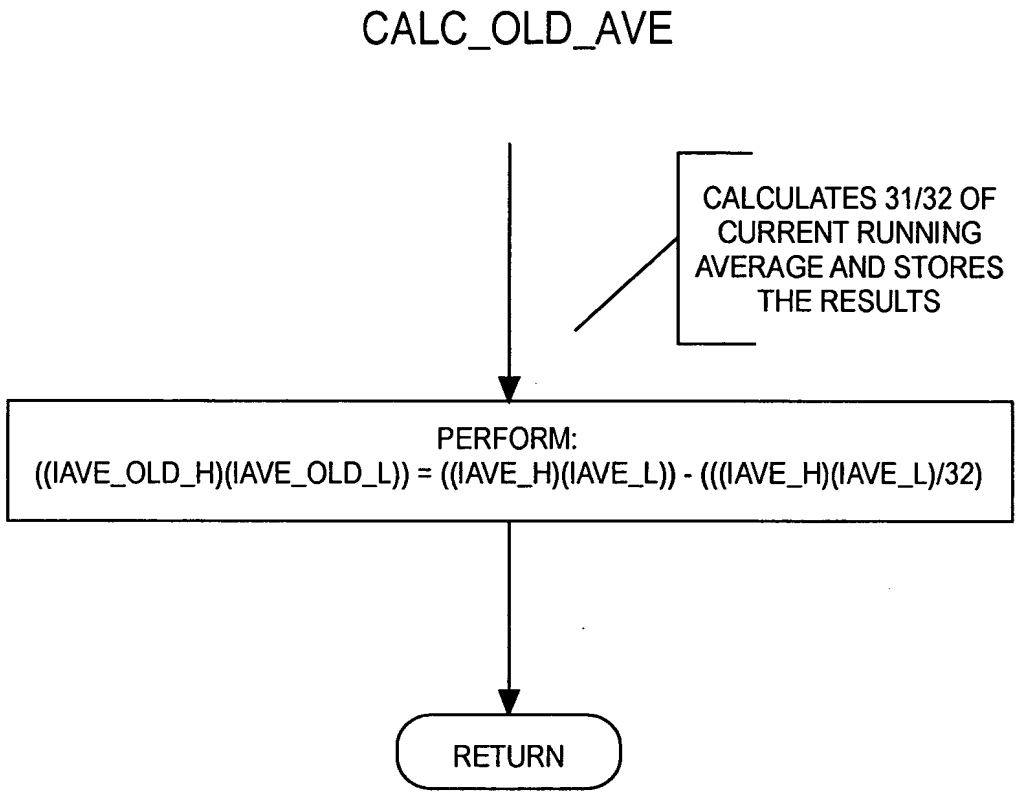

CALC_OLD_AVE
CALCULATES 31/32 OF CURRENT RUNNING AVERAGE AND STORES THE RESULTS
PERFORM:
((IAVE_OLD_H)(IAVE_OLD_L)) = ((IAVE_H)(IAVE_L)) - (((IAVE_H)(IAVE_L)/32)
RETURN

Fig. 12

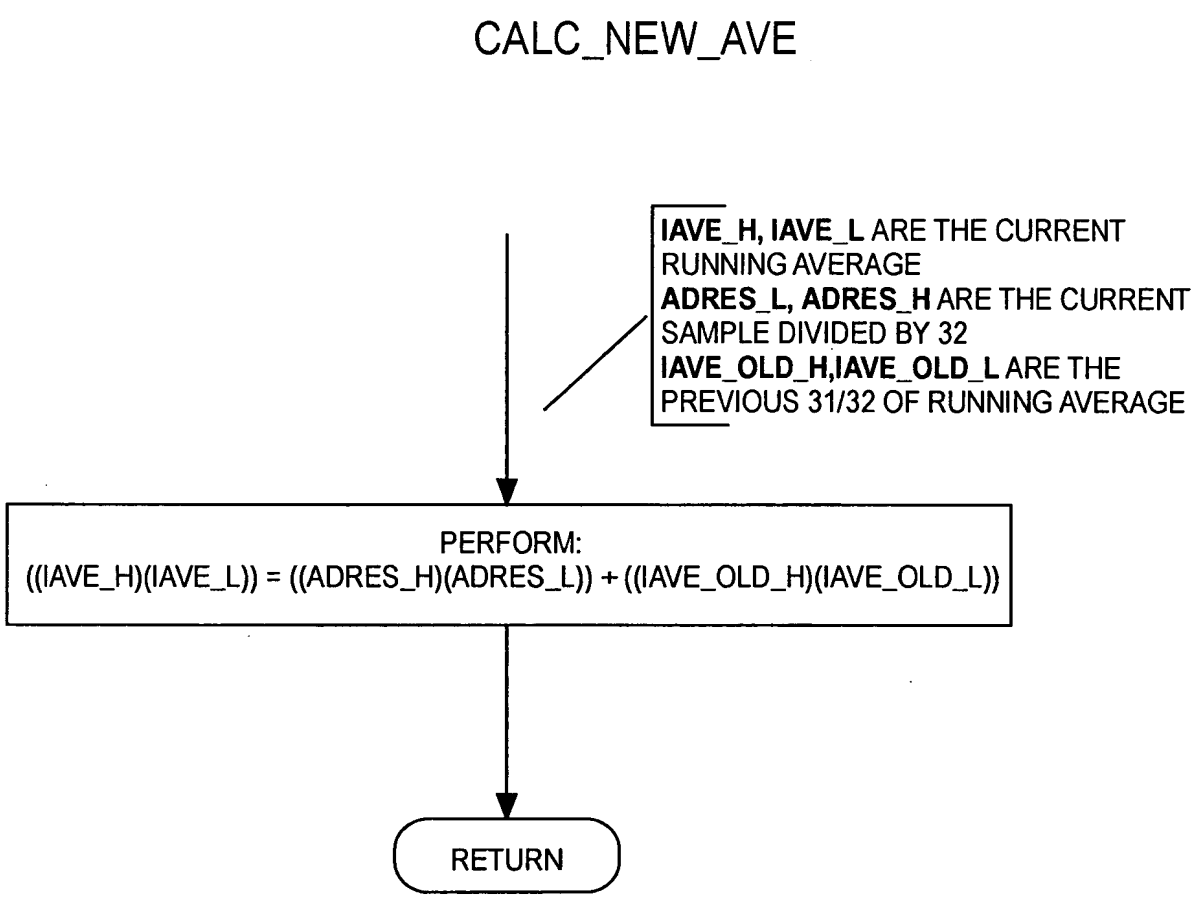
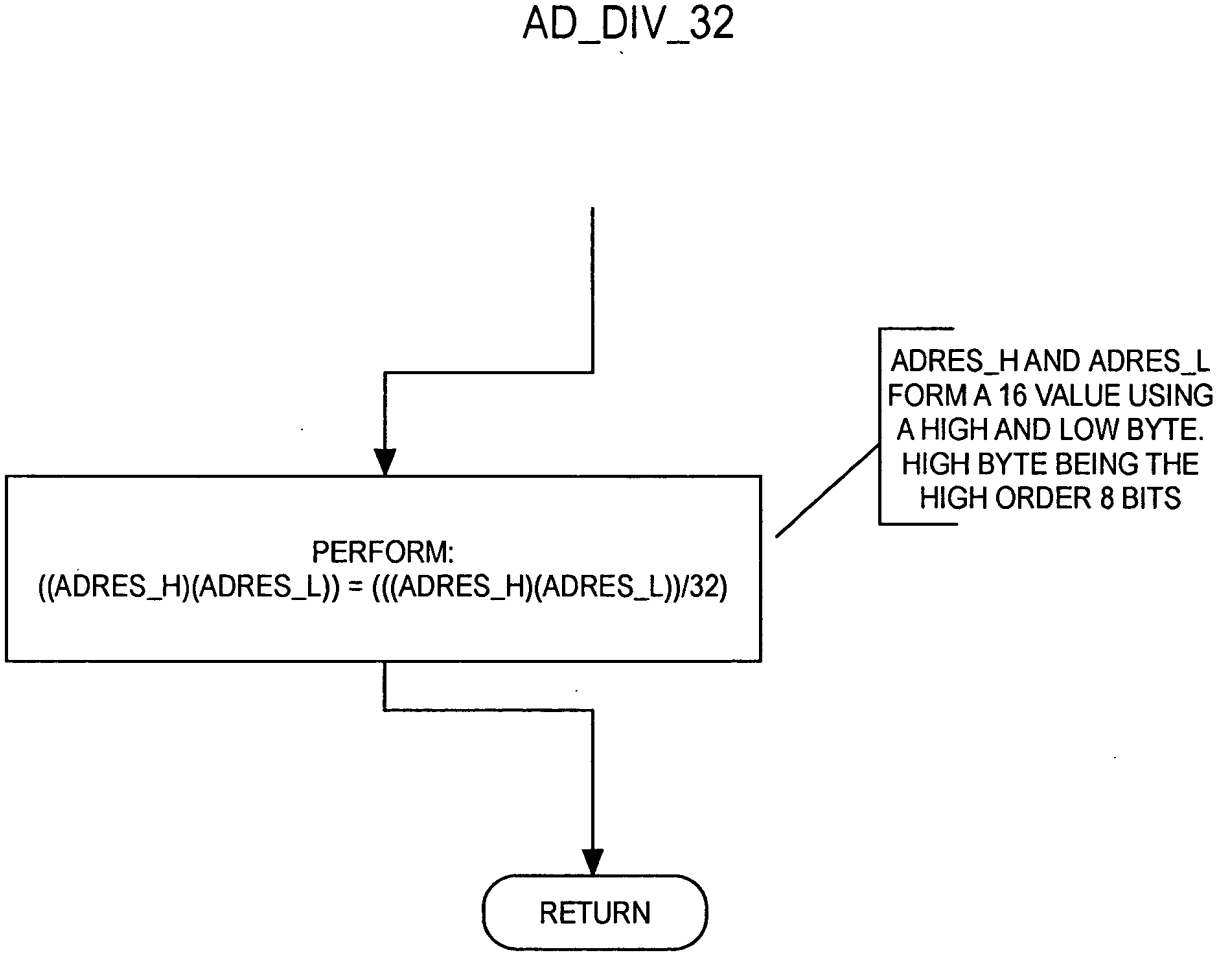

CALC_NEW_AVE
IAVE_H, IAVE_L ARE THE CURRENT RUNNING AVERAGE
ADRES_L, ADRES_H ARE THE CURRENT SAMPLE DIVIDED BY 32
IAVE_OLD_H,IAVE_OLD_L ARE THE PREVIOUS 31/32 OF RUNNING AVERAGE
PERFORM:
((IAVE_H)(IAVE_L)) = ((ADRES_H)(ADRES_L)) + ((IAVE_OLD_H)(IAVE_OLD_L))
RETURN

Fig. 13


AD_DIV_32
ADRES_H AND ADRES_L FORM A 16 VALUE USING A HIGH AND LOW BYTE. HIGH BYTE BEING THE HIGH ORDER 8 BITS
PERFORM:
((ADRES_H)(ADRES_L)) = (((ADRES_H)(ADRES_L))/32)
RETURN

PWMRAMPDOWN

Fig. 14

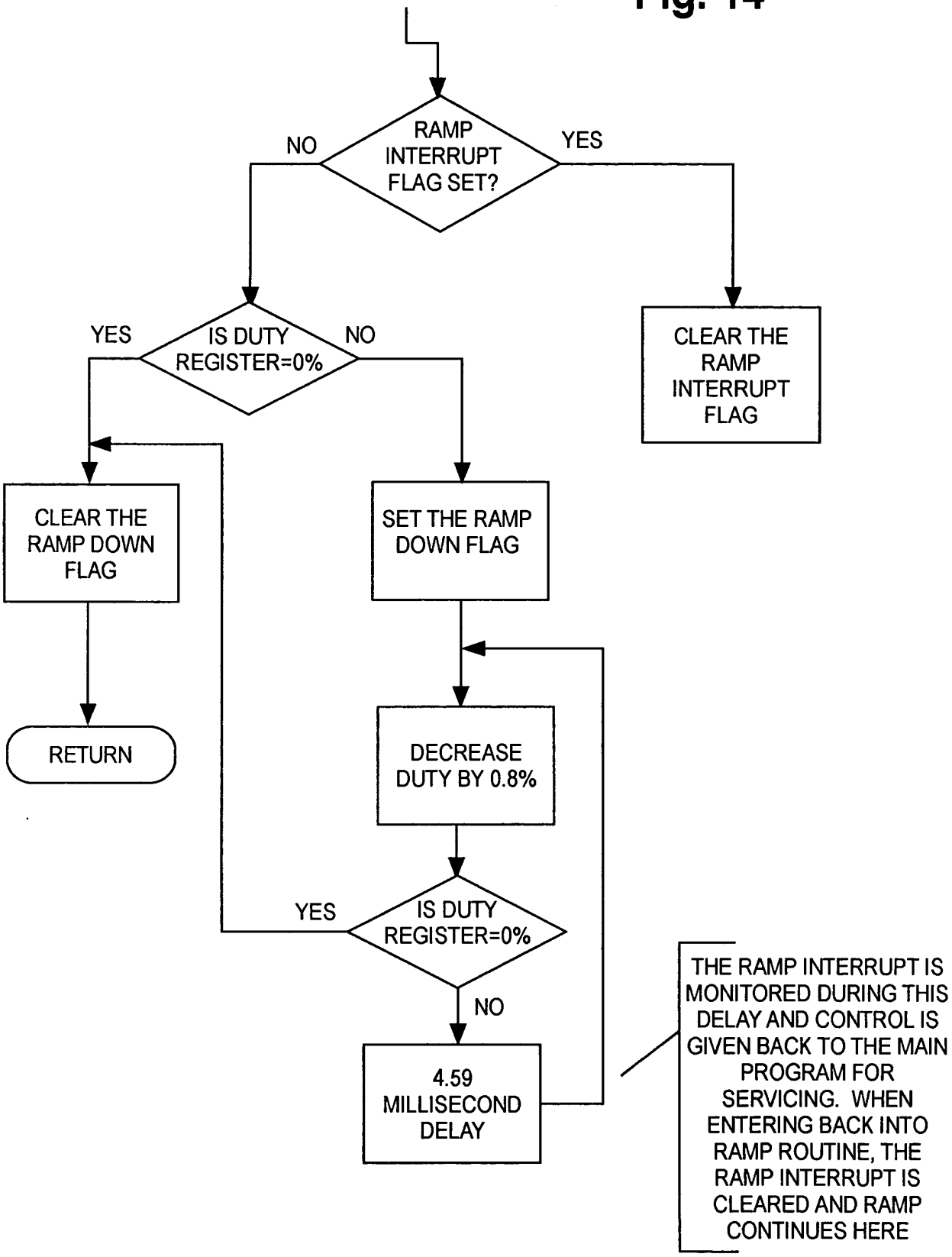


Fig. 15

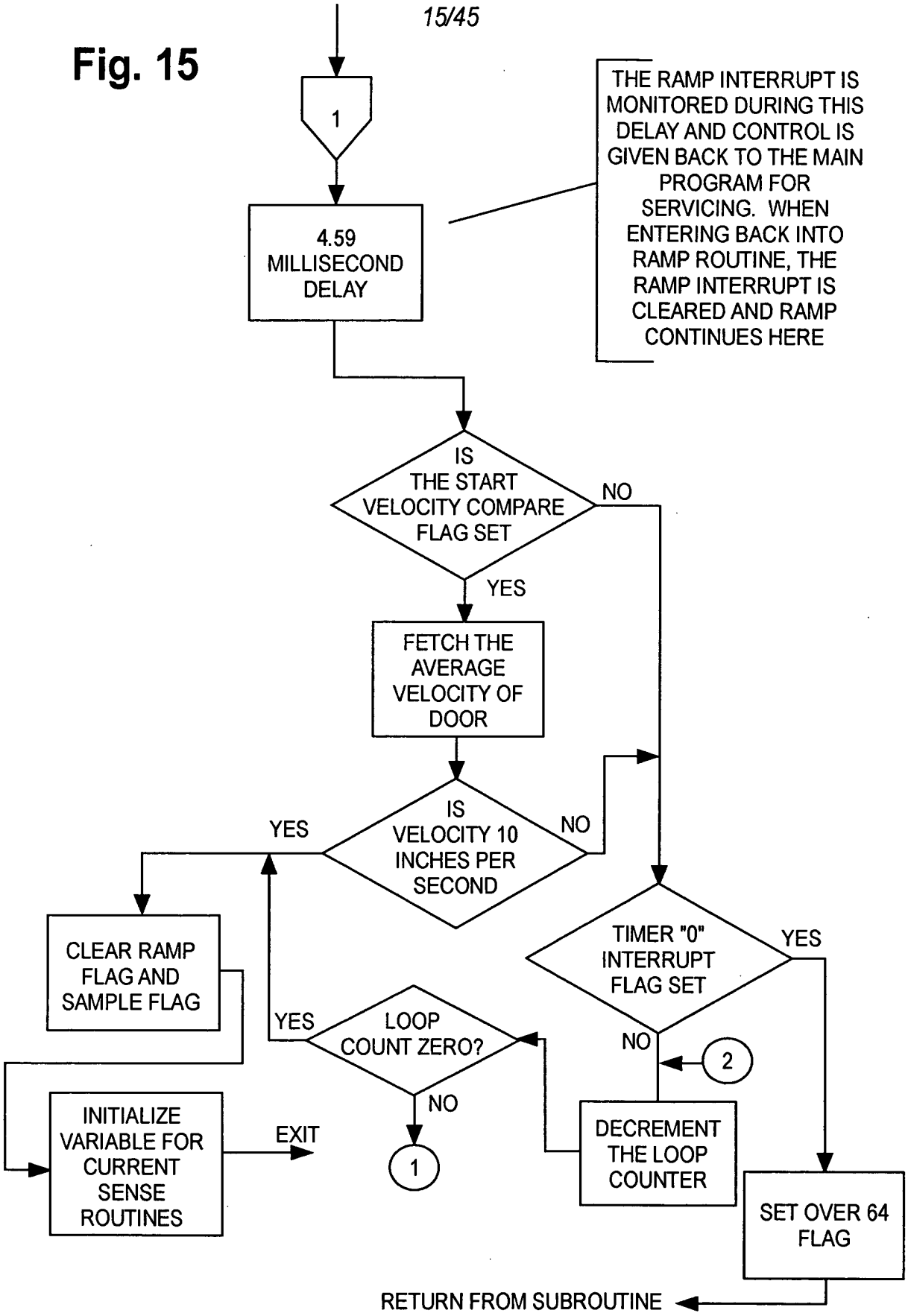

1
4.59 MILLISECOND DELAY
THE RAMP INTERRUPT IS MONITORED DURING THIS DELAY AND CONTROL IS GIVEN BACK TO THE MAIN PROGRAM FOR SERVICING. WHEN ENTERING BACK INTO RAMP ROUTINE, THE RAMP INTERRUPT IS CLEARED AND RAMP CONTINUES HERE
IS THE START VELOCITY COMPARE FLAG SET
NO
YES
FETCH THE AVERAGE VELOCITY OF DOOR
IS VELOCITY 10 INCHES PER SECOND
YES
NO
CLEAR RAMP FLAG AND SAMPLE FLAG
TIMER "0" INTERRUPT FLAG SET
YES
LOOP COUNT ZERO?
YES
NO
2
INITIALIZE VARIABLE FOR CURRENT SENSE ROUTINES
EXIT
NO
1
DECREMENT THE LOOP COUNTER
SET OVER 64 FLAG
RETURN FROM SUBROUTINE

Fig. 16

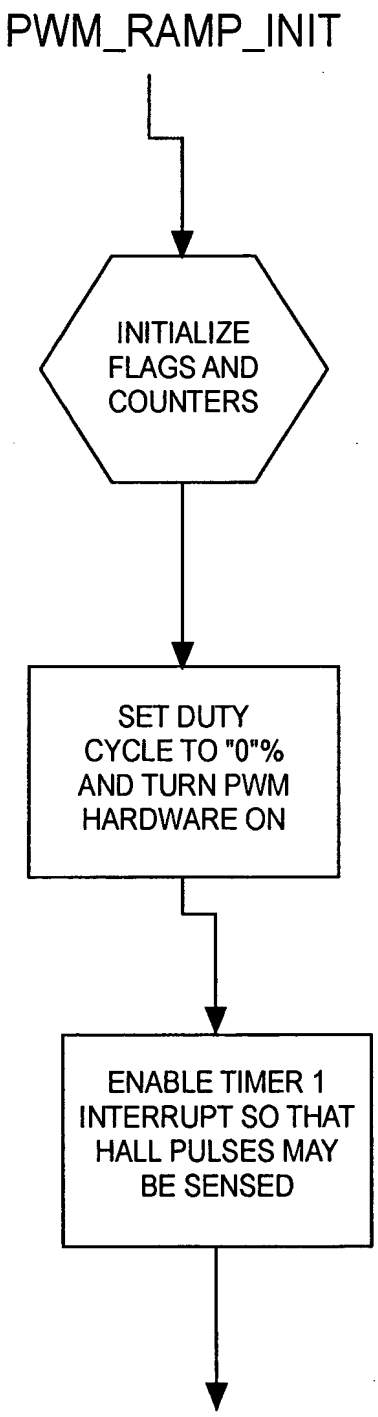

PWM_RAMP_INIT
INITIALIZE FLAGS AND COUNTERS
SET DUTY CYCLE TO "0"% AND TURN PWM HARDWARE ON
ENABLE TIMER 1 INTERRUPT SO THAT HALL PULSES MAY BE SENSED

Fig. 17

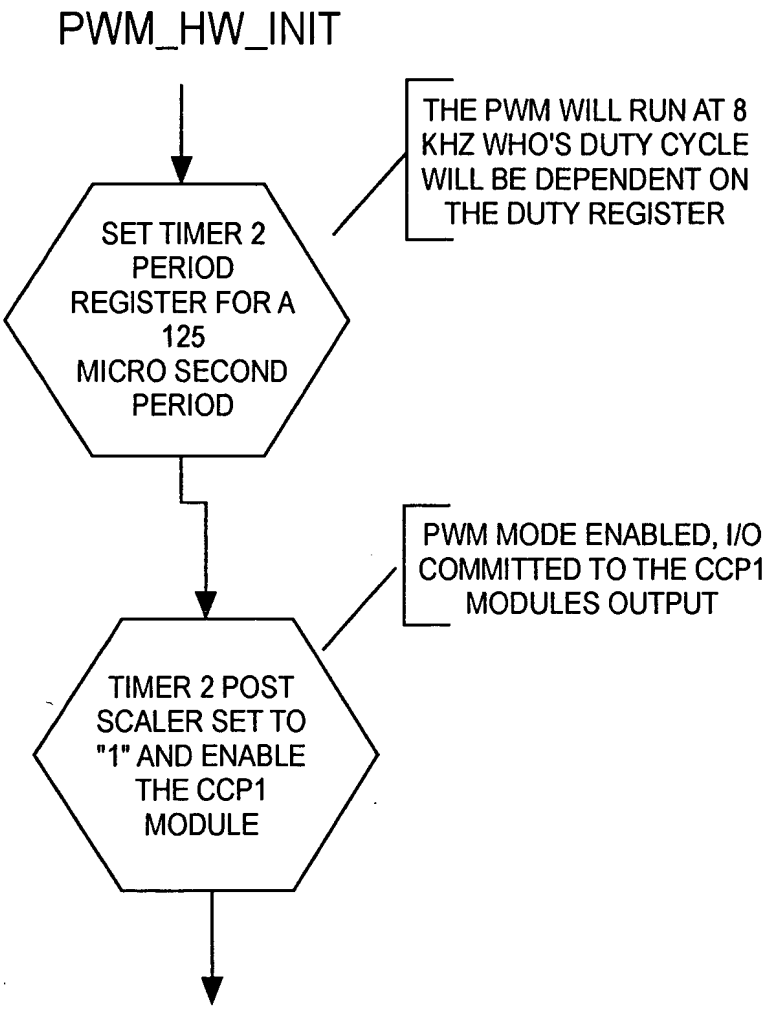

PWM_HW_INIT
SET TIMER 2 PERIOD REGISTER FOR A 125 MICRO SECOND PERIOD
THE PWM WILL RUN AT 8 KHZ WHO'S DUTY CYCLE WILL BE DEPENDENT ON THE DUTY REGISTER
TIMER 2 POST SCALER SET TO "1" AND ENABLE THE CCP1 MODULE
PWM MODE ENABLED, I/O COMMITTED TO THE CCP1 MODULES OUTPUT

Fig. 18

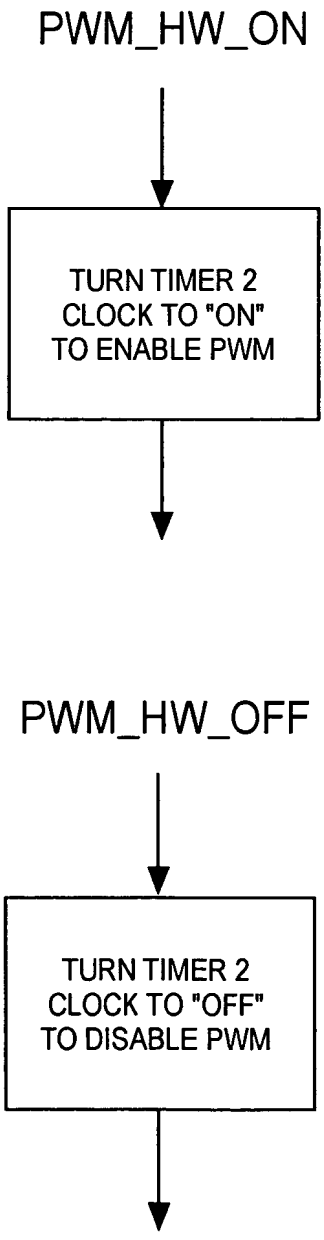

PWM_HW_ON
TURN TIMER 2
CLOCK TO "ON"
TO ENABLE PWM
PWM_HW_OFF
TURN TIMER 2
CLOCK TO "OFF"
TO DISABLE PWM

Fig. 19

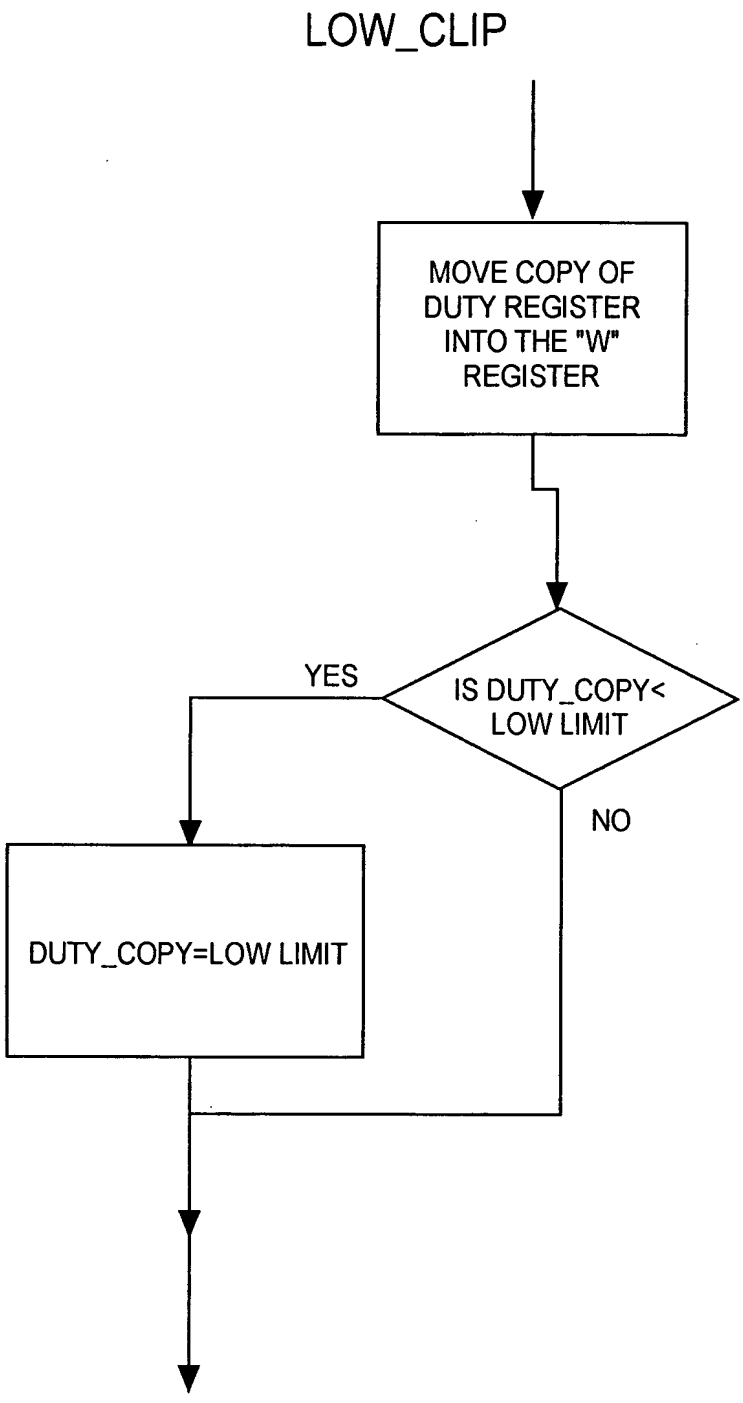

LOW_CLIP
MOVE COPY OF DUTY REGISTER INTO THE "W" REGISTER
IS DUTY_COPY< LOW LIMIT
YES
NO
DUTY_COPY=LOW LIMIT

Fig. 20

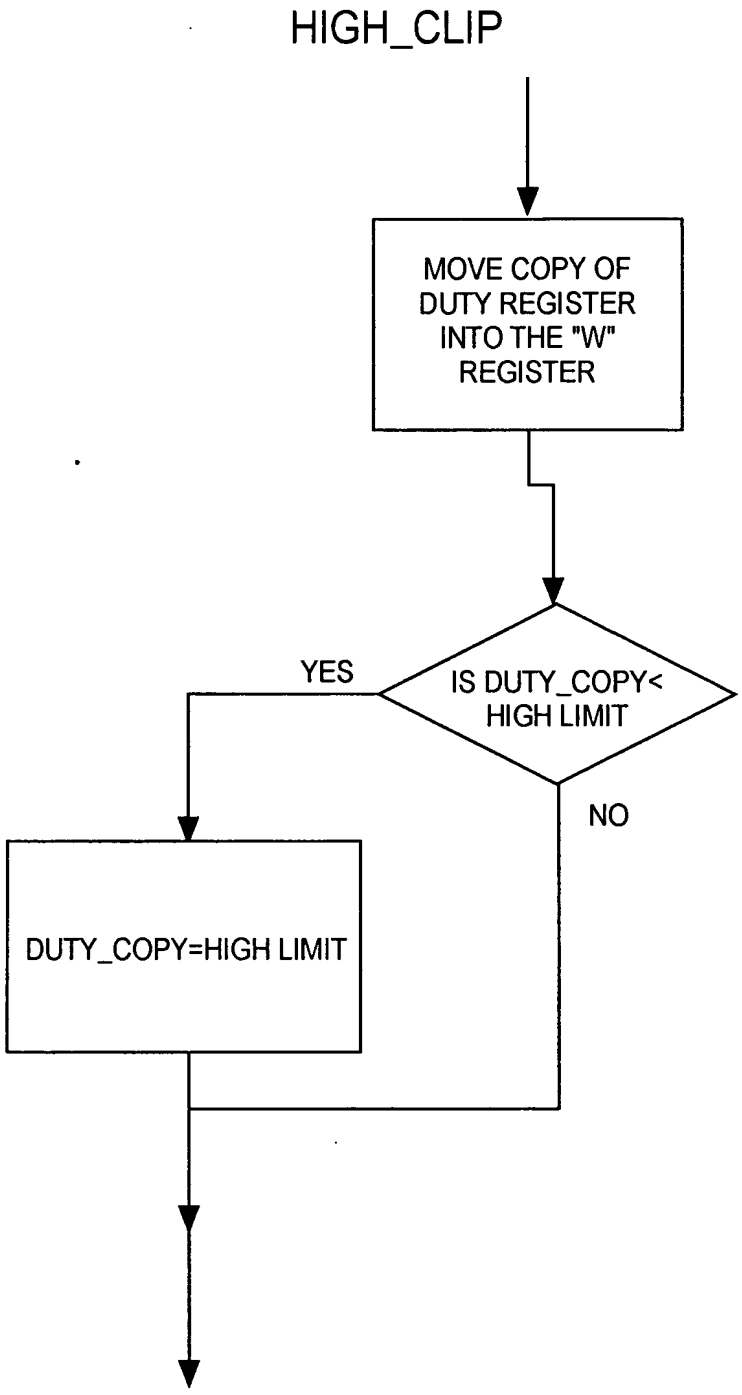

HIGH_CLIP
MOVE COPY OF DUTY REGISTER INTO THE "W" REGISTER
IS DUTY_COPY< HIGH LIMIT
YES
NO
DUTY_COPY=HIGH LIMIT

Fig. 21

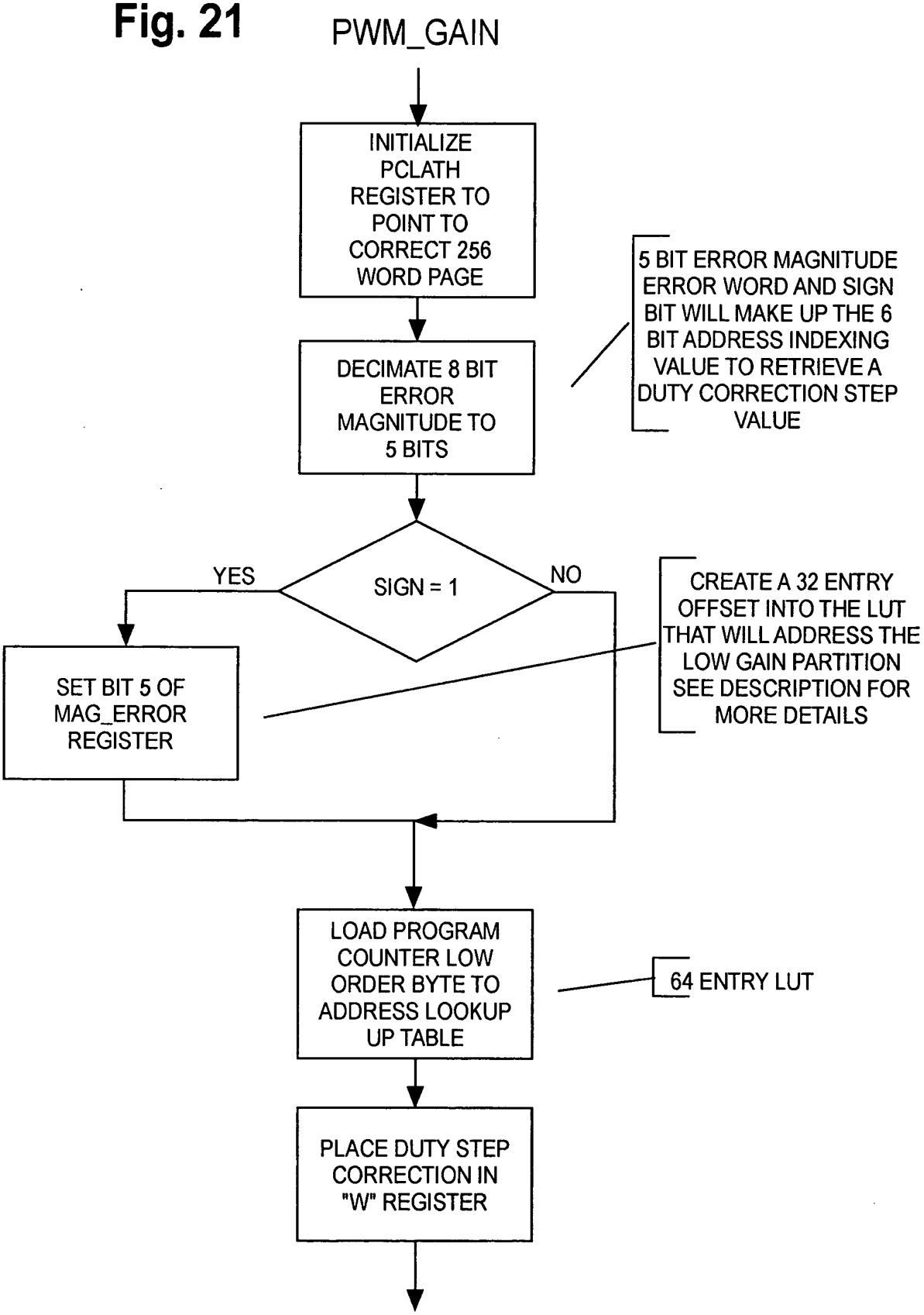
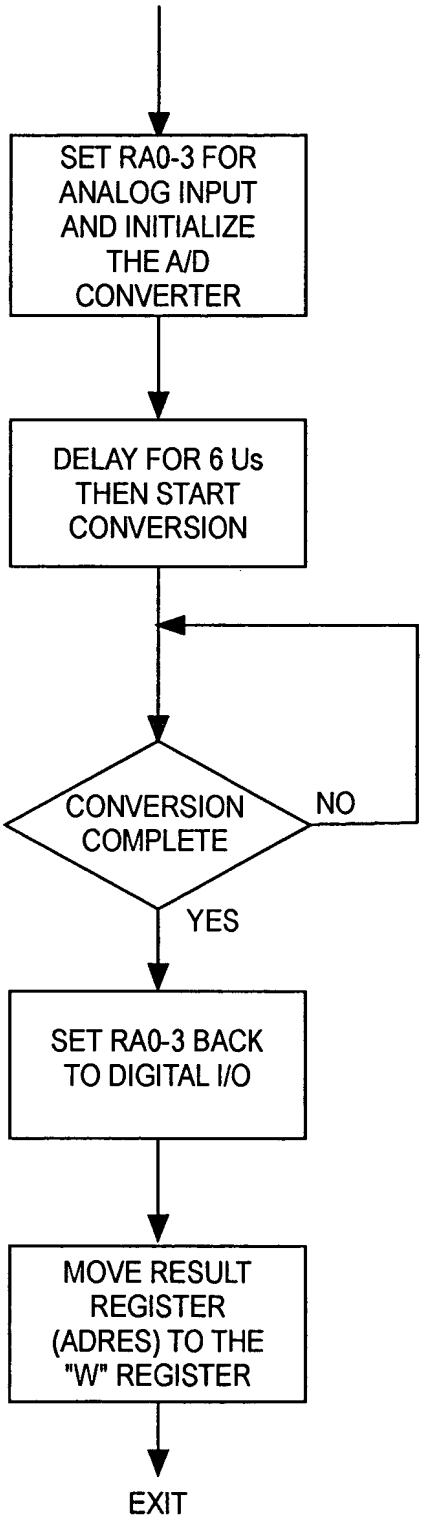

PWM_GAIN
INITIALIZE PCLATH REGISTER TO POINT TO CORRECT 256 WORD PAGE
DECIMATE 8 BIT ERROR MAGNITUDE TO 5 BITS
5 BIT ERROR MAGNITUDE ERROR WORD AND SIGN BIT WILL MAKE UP THE 6 BIT ADDRESS INDEXING VALUE TO RETRIEVE A DUTY CORRECTION STEP VALUE
SIGN = 1
YES
NO
SET BIT 5 OF MAG_ERROR REGISTER
CREATE A 32 ENTRY OFFSET INTO THE LUT THAT WILL ADDRESS THE LOW GAIN PARTITION SEE DESCRIPTION FOR MORE DETAILS
LOAD PROGRAM COUNTER LOW ORDER BYTE TO ADDRESS LOOKUP UP TABLE
64 ENTRY LUT
PLACE DUTY STEP CORRECTION IN "W" REGISTER

Fig. 22


SET RA0-3 FOR ANALOG INPUT AND INITIALIZE THE A/D CONVERTER
DELAY FOR 6 Us THEN START CONVERSION
CONVERSION COMPLETE
NO
YES
SET RA0-3 BACK TO DIGITAL I/O
MOVE RESULT REGISTER (ADRES) TO THE "W" REGISTER
EXIT

PWM

Fig. 23

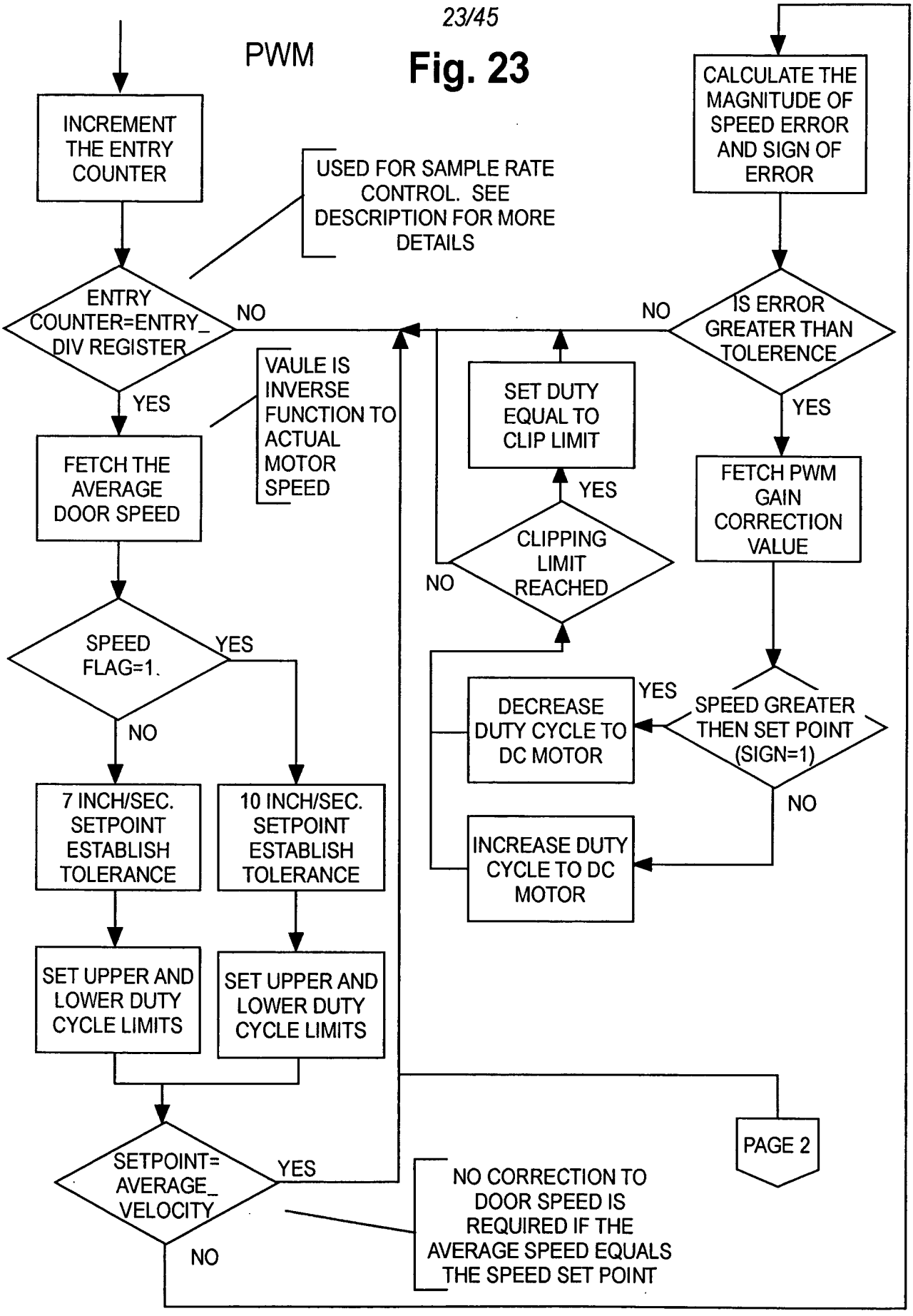

INCREMENT THE ENTRY COUNTER
USED FOR SAMPLE RATE CONTROL. SEE DESCRIPTION FOR MORE DETAILS
ENTRY COUNTER=ENTRY_ DIV REGISTER
NO
YES
VAULE IS INVERSE FUNCTION TO ACTUAL MOTOR SPEED
FETCH THE AVERAGE DOOR SPEED
SPEED FLAG=1.
YES
NO
7 INCH/SEC. SETPOINT ESTABLISH TOLERANCE
10 INCH/SEC. SETPOINT ESTABLISH TOLERANCE
SET UPPER AND LOWER DUTY CYCLE LIMITS
SET UPPER AND LOWER DUTY CYCLE LIMITS
SETPOINT= AVERAGE_ VELOCITY
YES
NO
NO CORRECTION TO DOOR SPEED IS REQUIRED IF THE AVERAGE SPEED EQUALS THE SPEED SET POINT
CALCULATE THE MAGNITUDE OF SPEED ERROR AND SIGN OF ERROR
IS ERROR GREATER THAN TOLERENCE
NO
YES
FETCH PWM GAIN CORRECTION VALUE
SPEED GREATER THEN SET POINT (SIGN=1)
YES
NO
DECREASE DUTY CYCLE TO DC MOTOR
INCREASE DUTY CYCLE TO DC MOTOR
SET DUTY EQUAL TO CLIP LIMIT
YES
CLIPPING LIMIT REACHED
NO
PAGE 2

Fig. 24

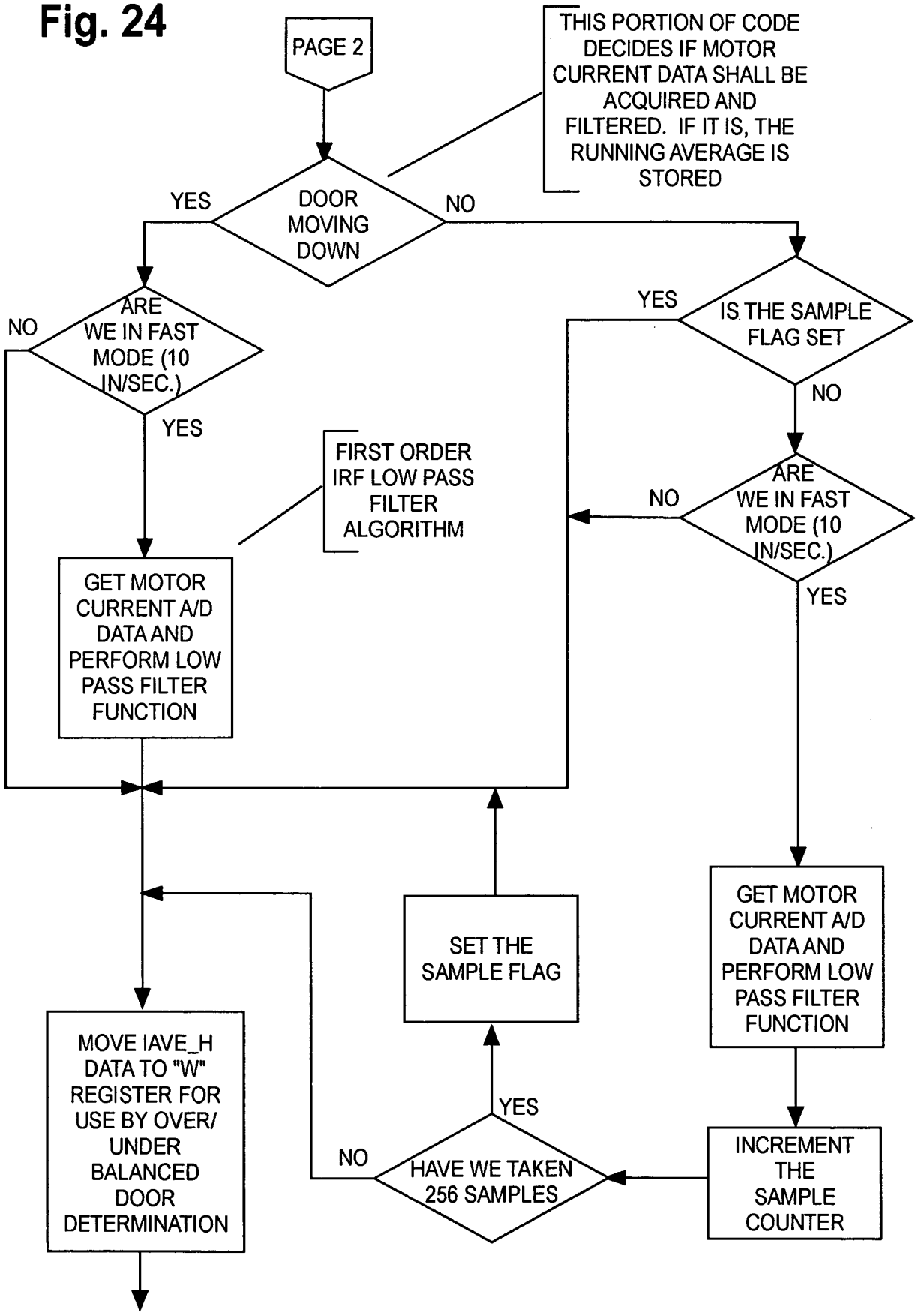

PAGE 2
THIS PORTION OF CODE DECIDES IF MOTOR CURRENT DATA SHALL BE ACQUIRED AND FILTERED. IF IT IS, THE RUNNING AVERAGE IS STORED
DOOR MOVING DOWN
YES
NO
ARE WE IN FAST MODE (10 IN/SEC.)
NO
YES
IS THE SAMPLE FLAG SET
YES
NO
ARE WE IN FAST MODE (10 IN/SEC.)
NO
YES
FIRST ORDER IRF LOW PASS FILTER ALGORITHM
GET MOTOR CURRENT A/D DATA AND PERFORM LOW PASS FILTER FUNCTION
GET MOTOR CURRENT A/D DATA AND PERFORM LOW PASS FILTER FUNCTION
SET THE SAMPLE FLAG
INCREMENT THE SAMPLE COUNTER
YES
NO
HAVE WE TAKEN 256 SAMPLES
MOVE IAVE_H DATA TO "W" REGISTER FOR USE BY OVER/ UNDER BALANCED DOOR DETERMINATION

Fig. 25

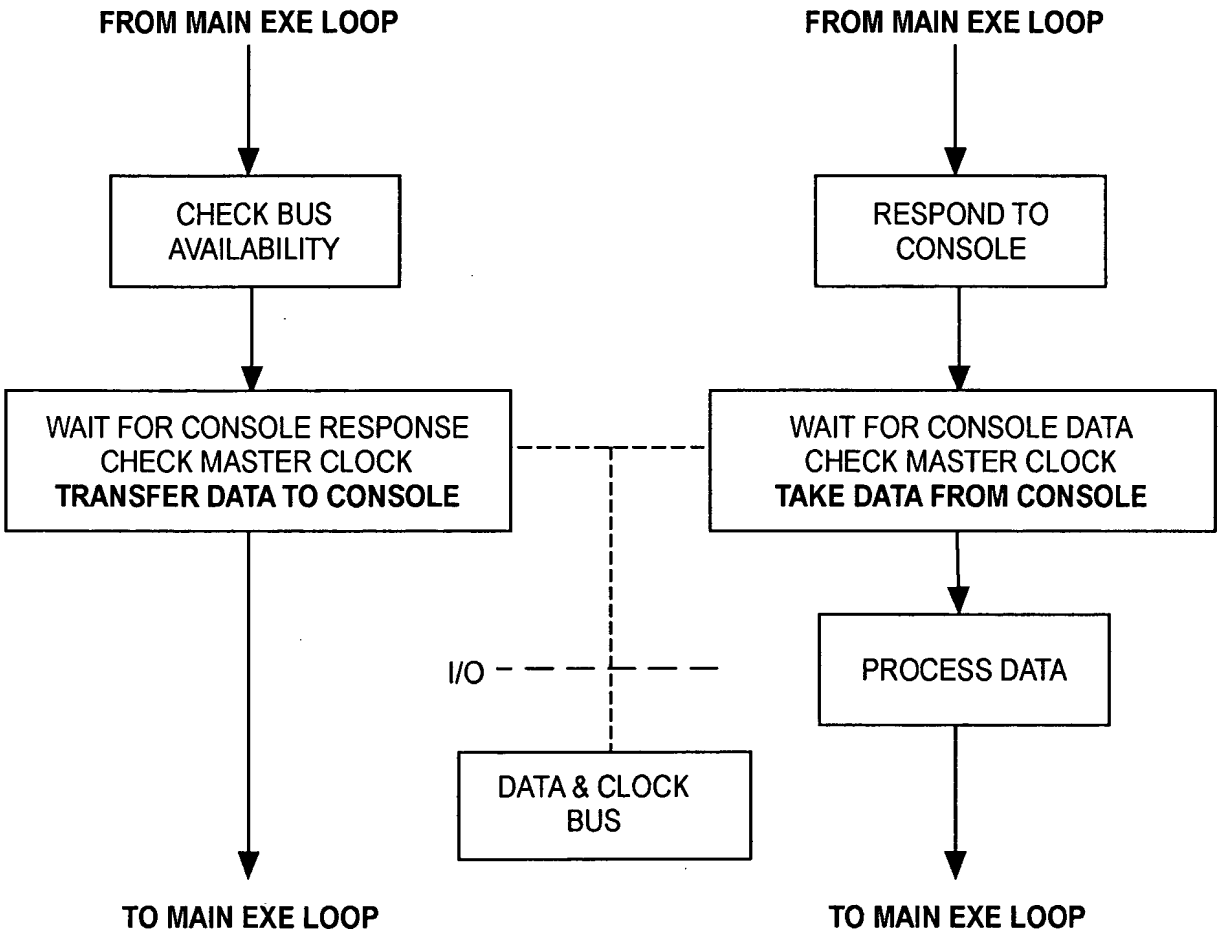

FROM MAIN EXE LOOP
FROM MAIN EXE LOOP
CHECK BUS AVAILABILITY
RESPOND TO CONSOLE
WAIT FOR CONSOLE RESPONSE
CHECK MASTER CLOCK
TRANSFER DATA TO CONSOLE
WAIT FOR CONSOLE DATA
CHECK MASTER CLOCK
TAKE DATA FROM CONSOLE
I/O
PROCESS DATA
DATA & CLOCK BUS
TO MAIN EXE LOOP
TO MAIN EXE LOOP

Fig. 26

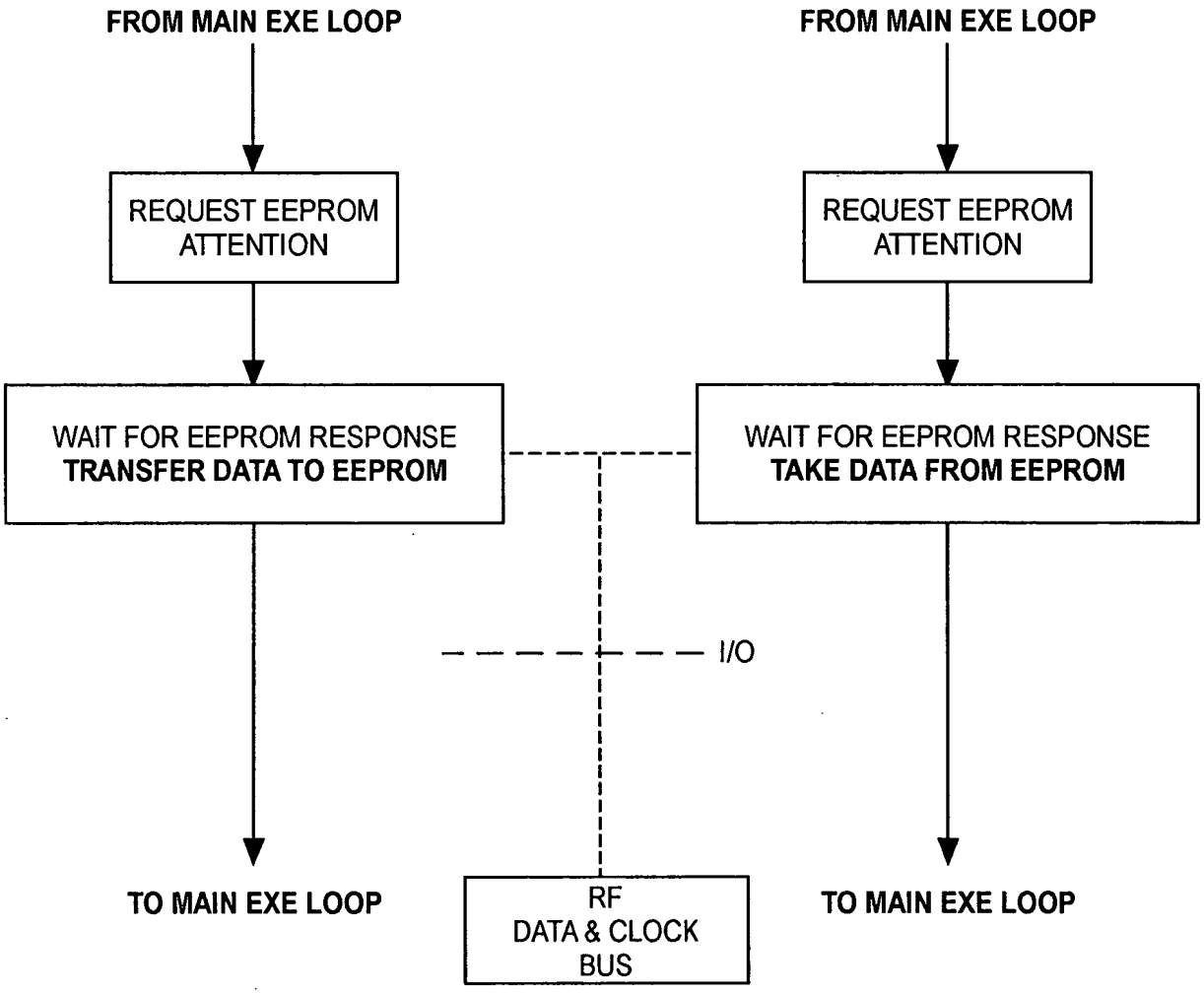

FROM MAIN EXE LOOP
FROM MAIN EXE LOOP
REQUEST EEPROM ATTENTION
REQUEST EEPROM ATTENTION
WAIT FOR EEPROM RESPONSE
TRANSFER DATA TO EEPROM
WAIT FOR EEPROM RESPONSE
TAKE DATA FROM EEPROM
I/O
TO MAIN EXE LOOP
RF
DATA & CLOCK
BUS
TO MAIN EXE LOOP

Fig. 27

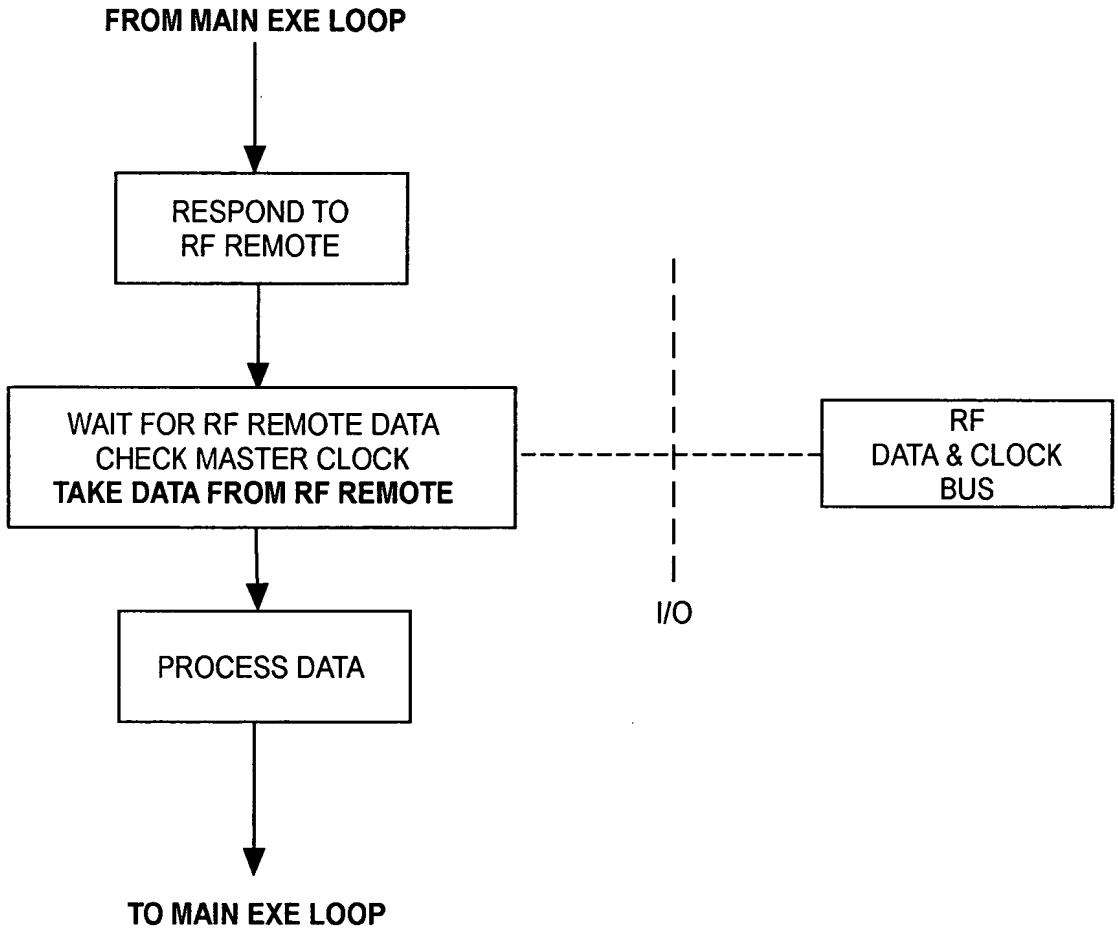

FROM MAIN EXE LOOP
RESPOND TO RF REMOTE
WAIT FOR RF REMOTE DATA
CHECK MASTER CLOCK
TAKE DATA FROM RF REMOTE
PROCESS DATA
TO MAIN EXE LOOP
I/O
RF DATA & CLOCK BUS

Fig. 28

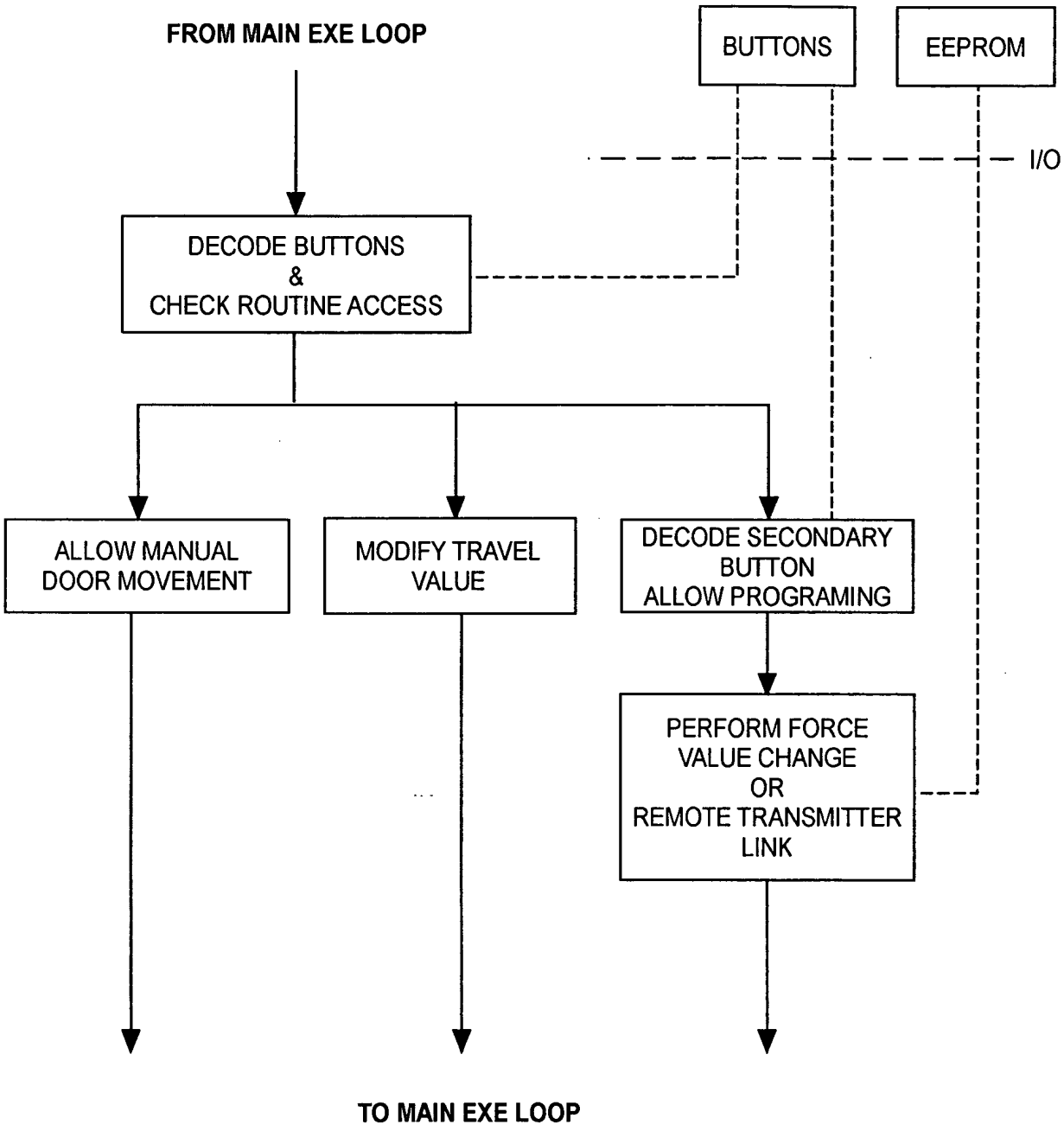

FROM MAIN EXE LOOP
BUTTONS
EEPROM
I/O
DECODE BUTTONS
&
CHECK ROUTINE ACCESS
ALLOW MANUAL
DOOR MOVEMENT
MODIFY TRAVEL
VALUE
DECODE SECONDARY
BUTTON
ALLOW PROGRAMING
PERFORM FORCE
VALUE CHANGE
OR
REMOTE TRANSMITTER
LINK
TO MAIN EXE LOOP

Fig. 29

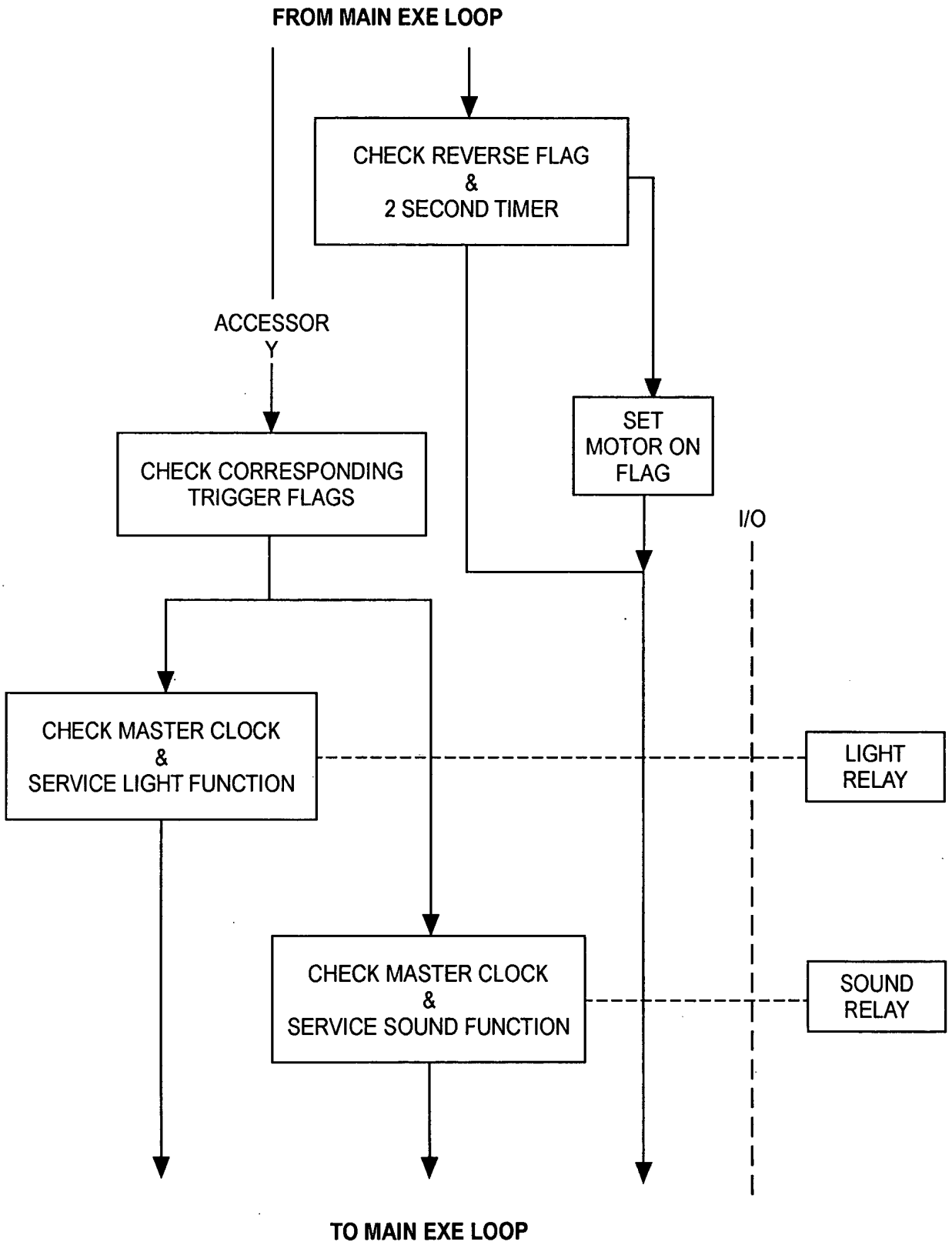

FROM MAIN EXE LOOP
CHECK REVERSE FLAG
&
2 SECOND TIMER
ACCESSOR
Y
SET
MOTOR ON
FLAG
CHECK CORRESPONDING
TRIGGER FLAGS
I/O
CHECK MASTER CLOCK
&
SERVICE LIGHT FUNCTION
LIGHT
RELAY
CHECK MASTER CLOCK
&
SERVICE SOUND FUNCTION
SOUND
RELAY
TO MAIN EXE LOOP

Fig. 30

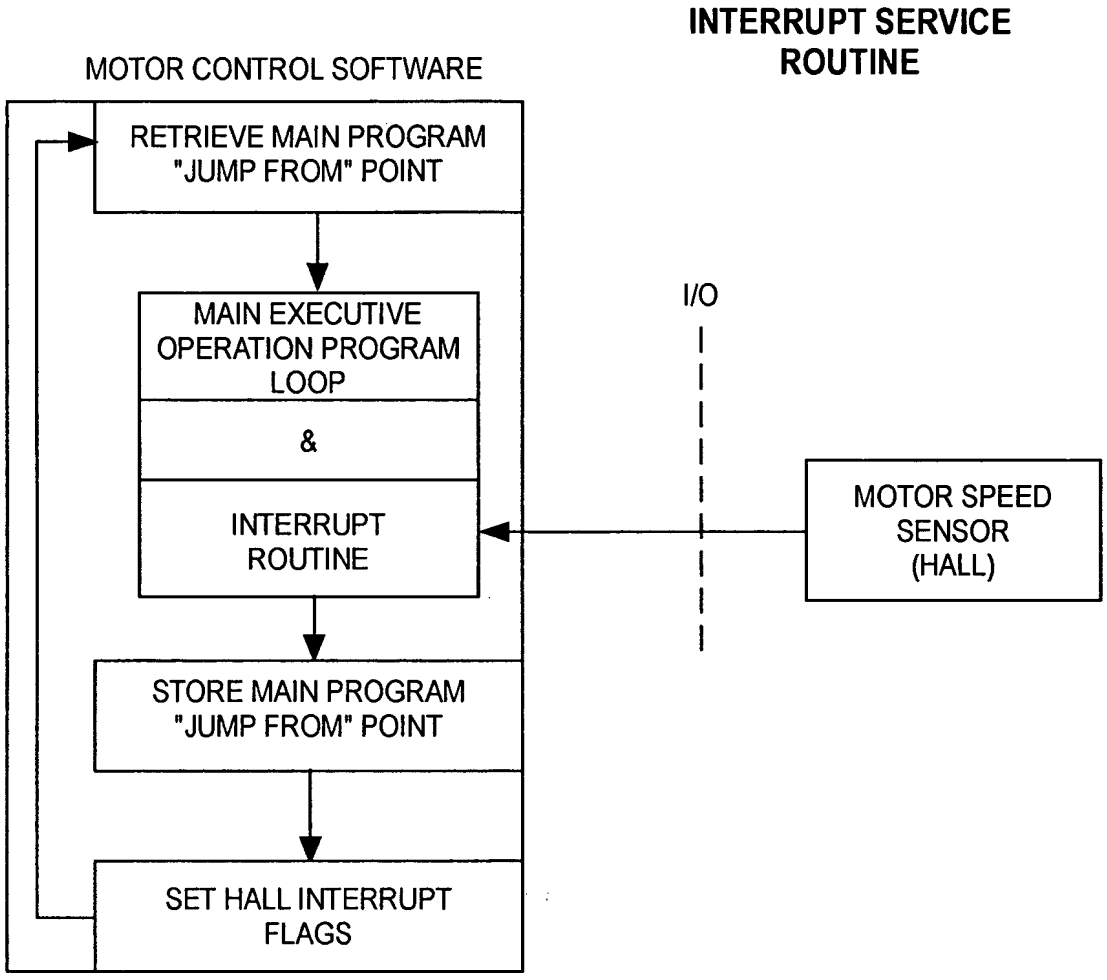

INTERRUPT SERVICE ROUTINE
MOTOR CONTROL SOFTWARE
RETRIEVE MAIN PROGRAM "JUMP FROM" POINT
MAIN EXECUTIVE OPERATION PROGRAM LOOP
&
INTERRUPT ROUTINE
STORE MAIN PROGRAM "JUMP FROM" POINT
SET HALL INTERRUPT FLAGS
I/O
MOTOR SPEED SENSOR (HALL)

Fig. 31

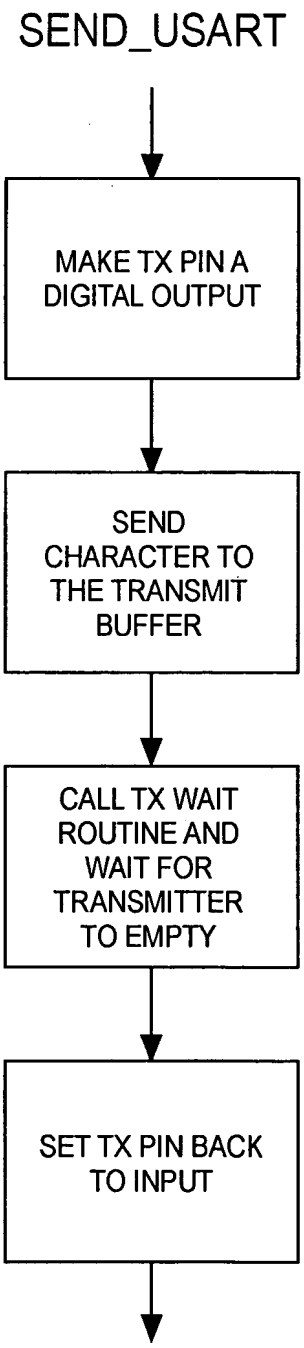

SEND_USART
MAKE TX PIN A DIGITAL OUTPUT
SEND CHARACTER TO THE TRANSMIT BUFFER
CALL TX WAIT ROUTINE AND WAIT FOR TRANSMITTER TO EMPTY
SET TX PIN BACK TO INPUT

Fig. 32

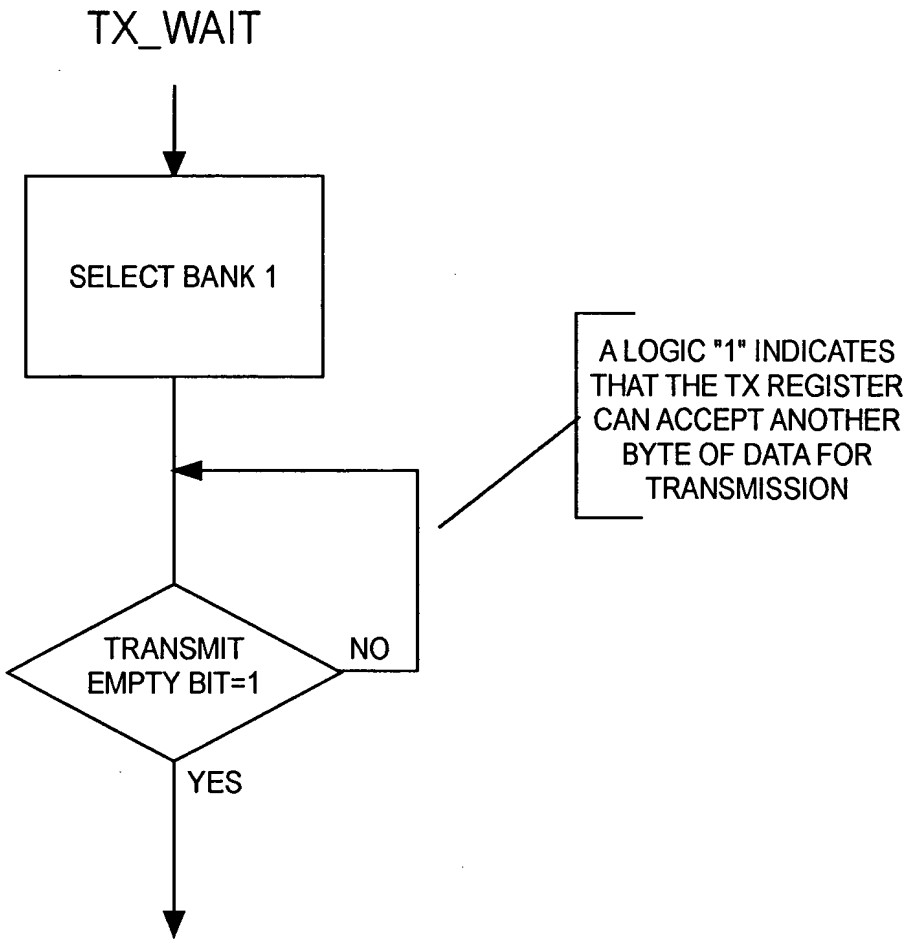

TX_WAIT
SELECT BANK 1
A LOGIC "1" INDICATES THAT THE TX REGISTER CAN ACCEPT ANOTHER BYTE OF DATA FOR TRANSMISSION
TRANSMIT EMPTY BIT=1
NO
YES

Fig. 33

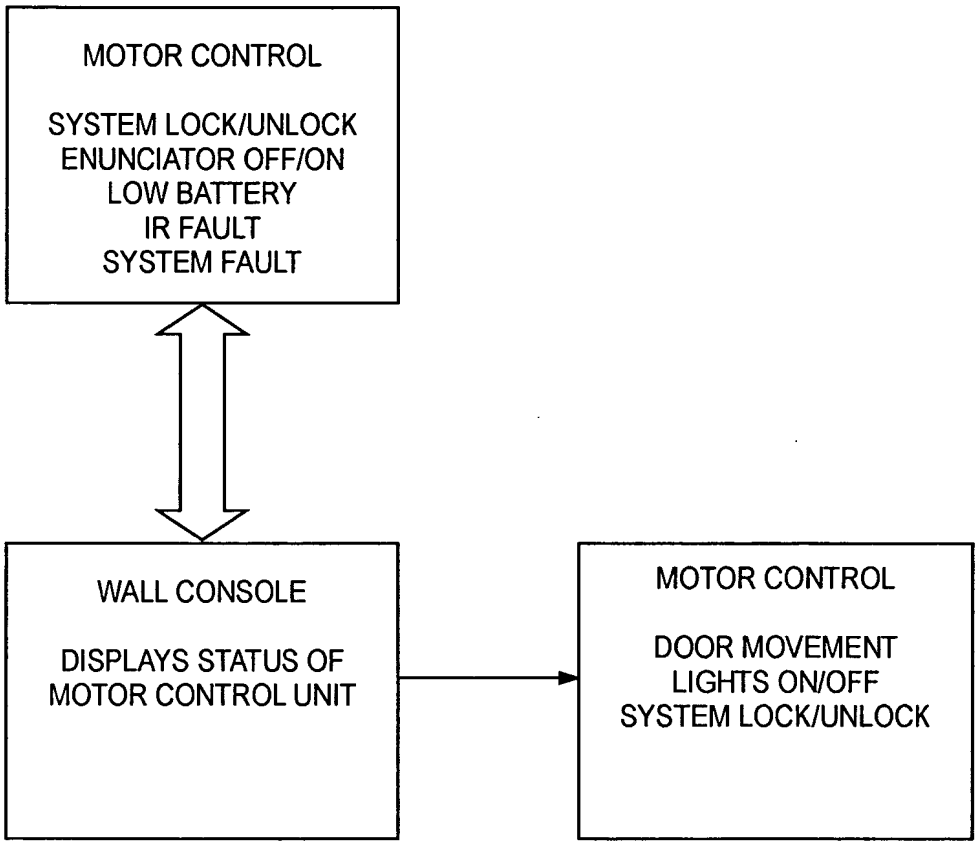

MOTOR CONTROL
SYSTEM LOCK/UNLOCK
ENUNCIATOR OFF/ON
LOW BATTERY
IR FAULT
SYSTEM FAULT
WALL CONSOLE
DISPLAYS STATUS OF
MOTOR CONTROL UNIT
MOTOR CONTROL
DOOR MOVEMENT
LIGHTS ON/OFF
SYSTEM LOCK/UNLOCK
WALL CONSOLE SYSTEM

Fig. 34

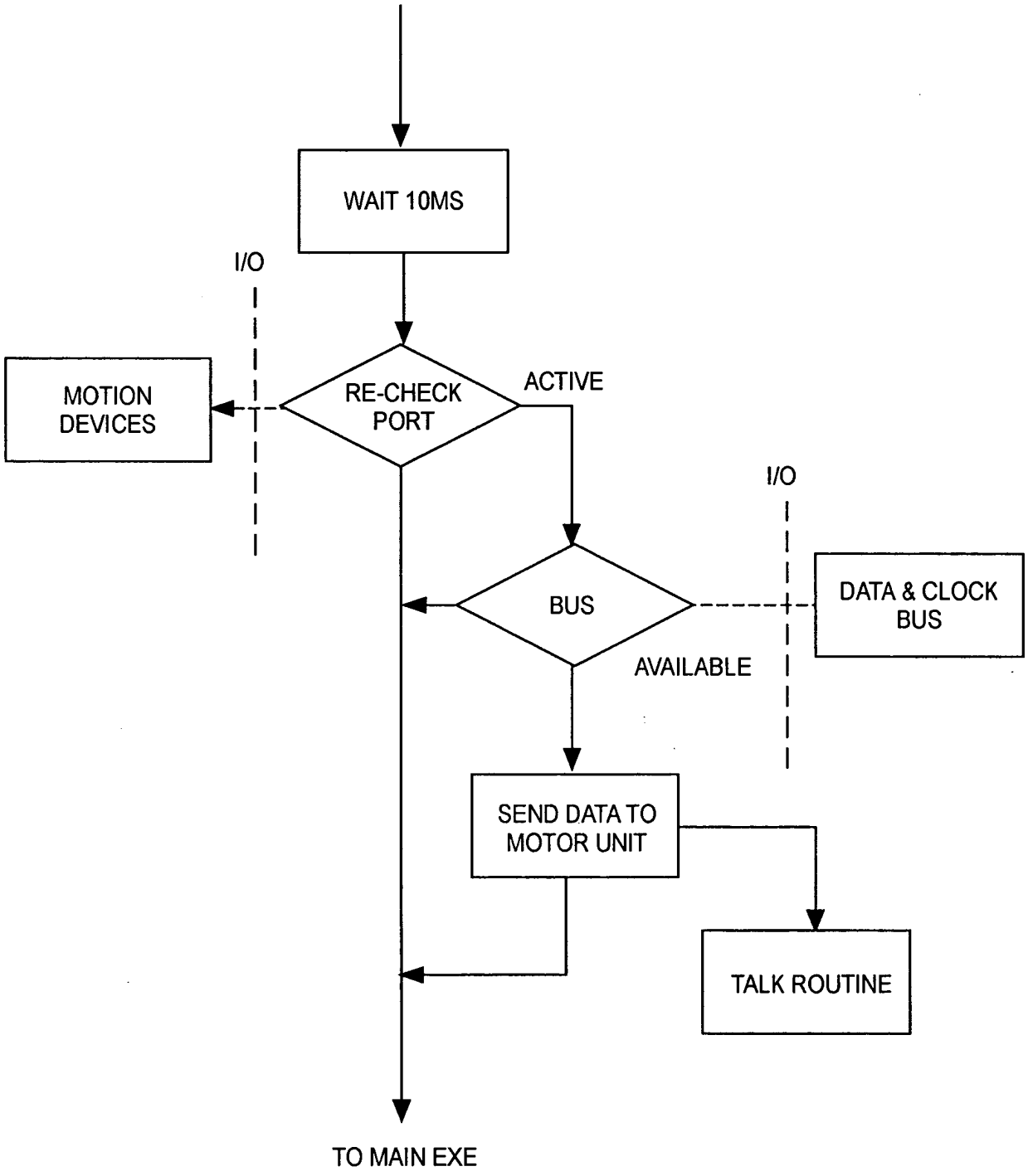

WAIT 10MS
I/O
MOTION DEVICES
RE-CHECK PORT
ACTIVE
I/O
BUS
DATA & CLOCK BUS
AVAILABLE
SEND DATA TO MOTOR UNIT
TALK ROUTINE
TO MAIN EXE

Fig. 35

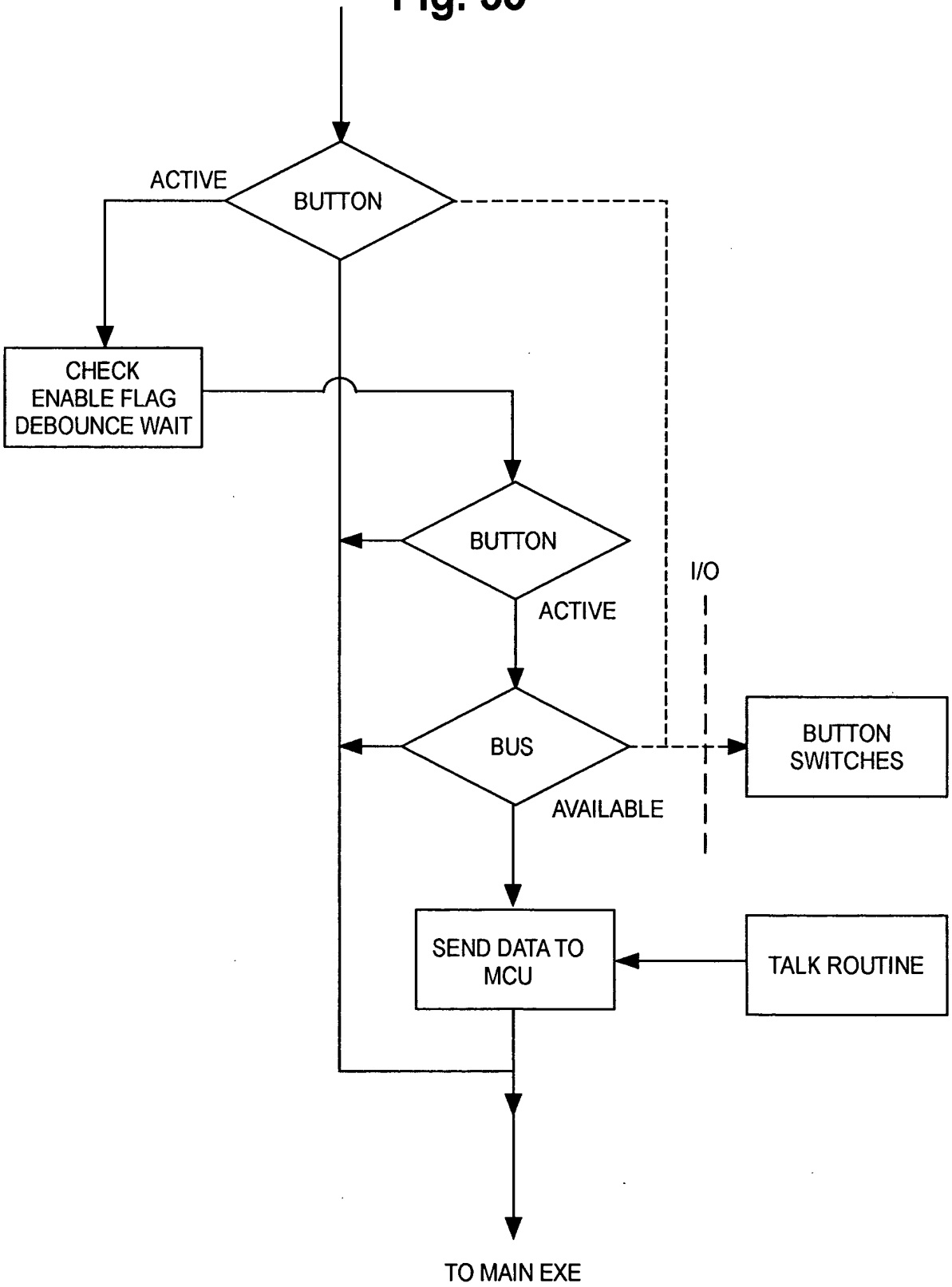

ACTIVE
BUTTON
CHECK
ENABLE FLAG
DEBOUNCE WAIT
BUTTON
I/O
ACTIVE
BUS
BUTTON
SWITCHES
AVAILABLE
SEND DATA TO
MCU
TALK ROUTINE
TO MAIN EXE

Fig. 36

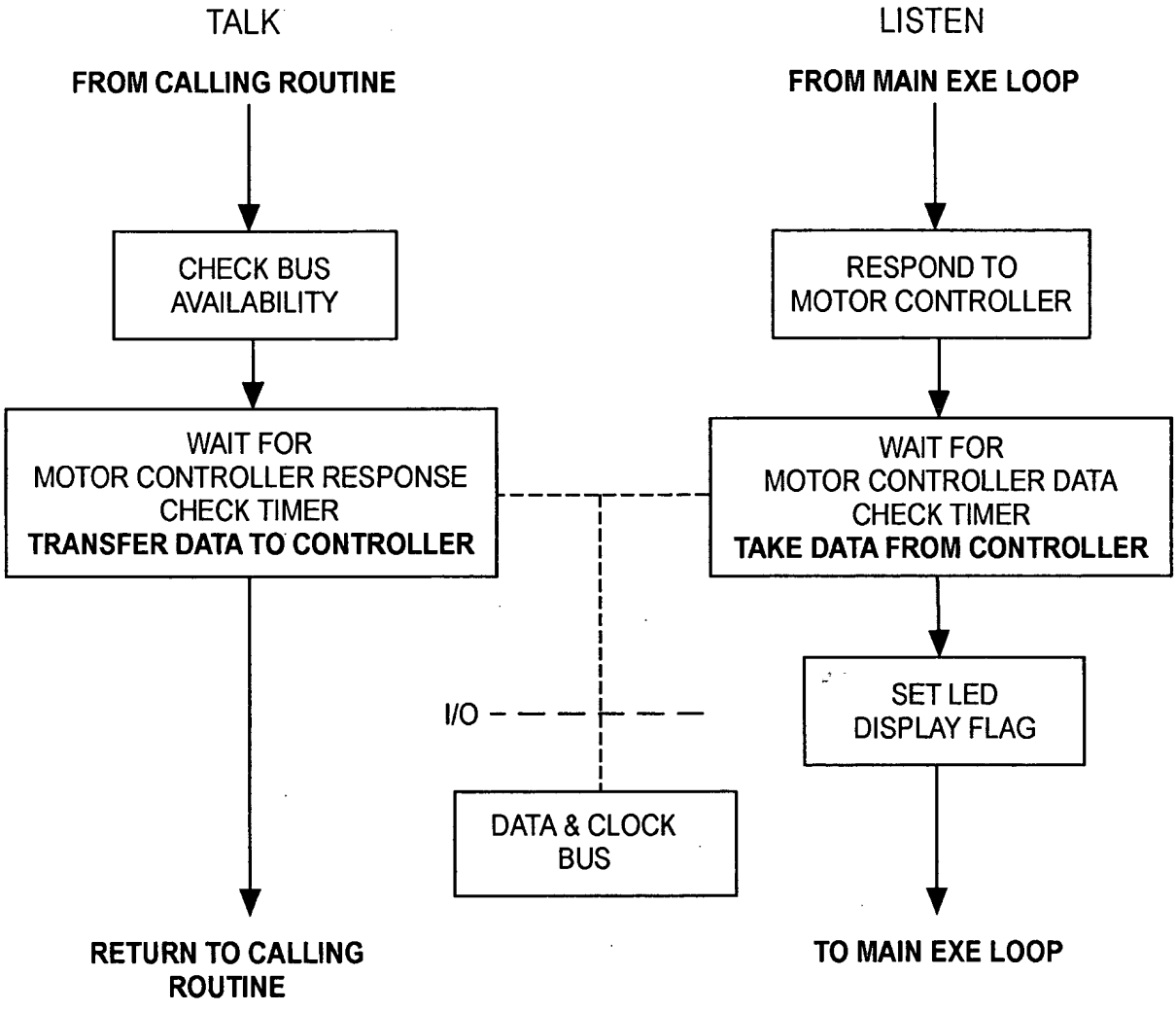

TALK
FROM CALLING ROUTINE
CHECK BUS AVAILABILITY
WAIT FOR MOTOR CONTROLLER RESPONSE CHECK TIMER
TRANSFER DATA TO CONTROLLER
RETURN TO CALLING ROUTINE
I/O
DATA & CLOCK BUS
LISTEN
FROM MAIN EXE LOOP
RESPOND TO MOTOR CONTROLLER
WAIT FOR MOTOR CONTROLLER DATA CHECK TIMER
TAKE DATA FROM CONTROLLER
SET LED DISPLAY FLAG
TO MAIN EXE LOOP

Fig. 37

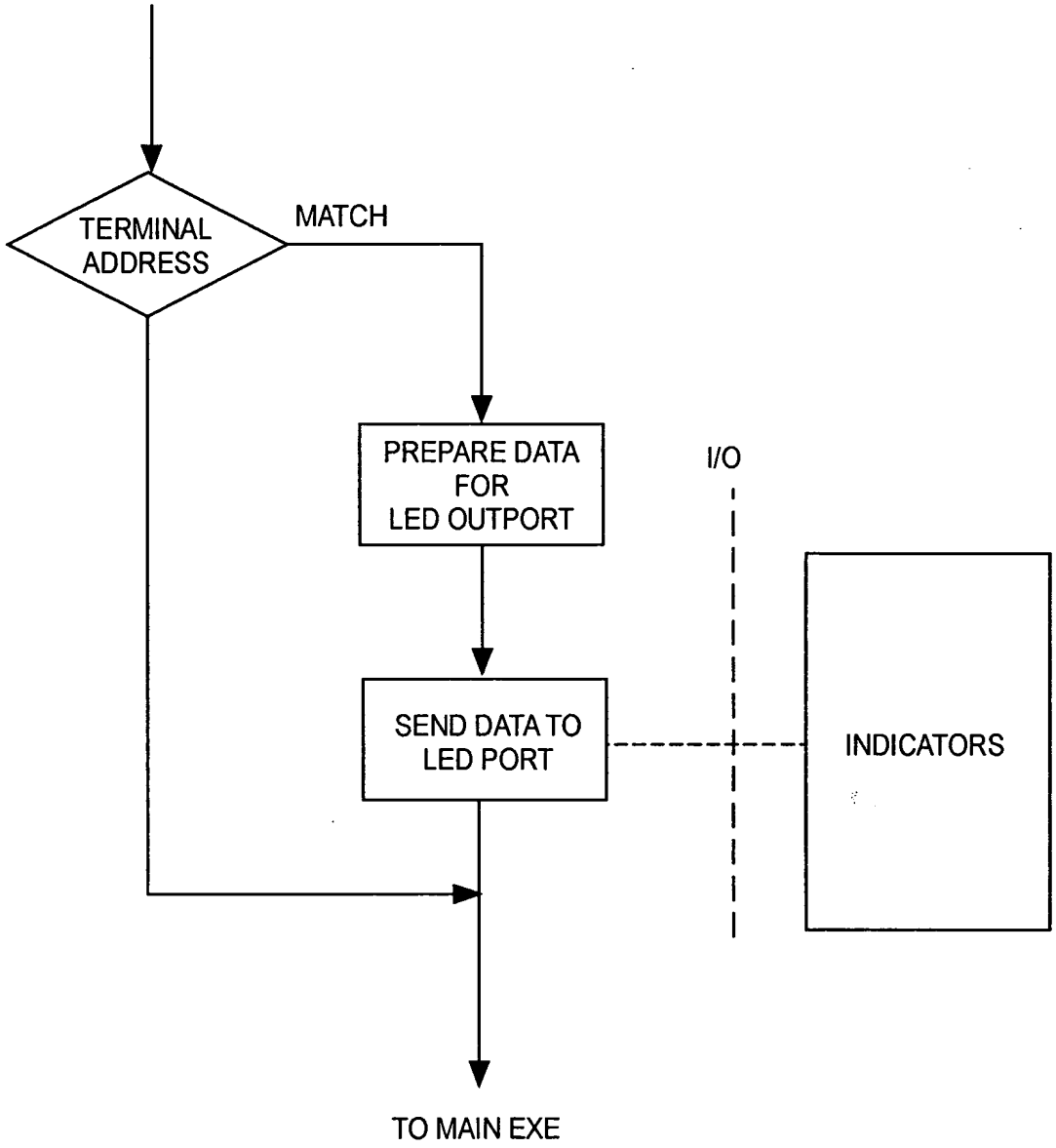

TERMINAL ADDRESS
MATCH
PREPARE DATA FOR LED OUTPORT
I/O
SEND DATA TO LED PORT
INDICATORS
TO MAIN EXE

Fig. 38

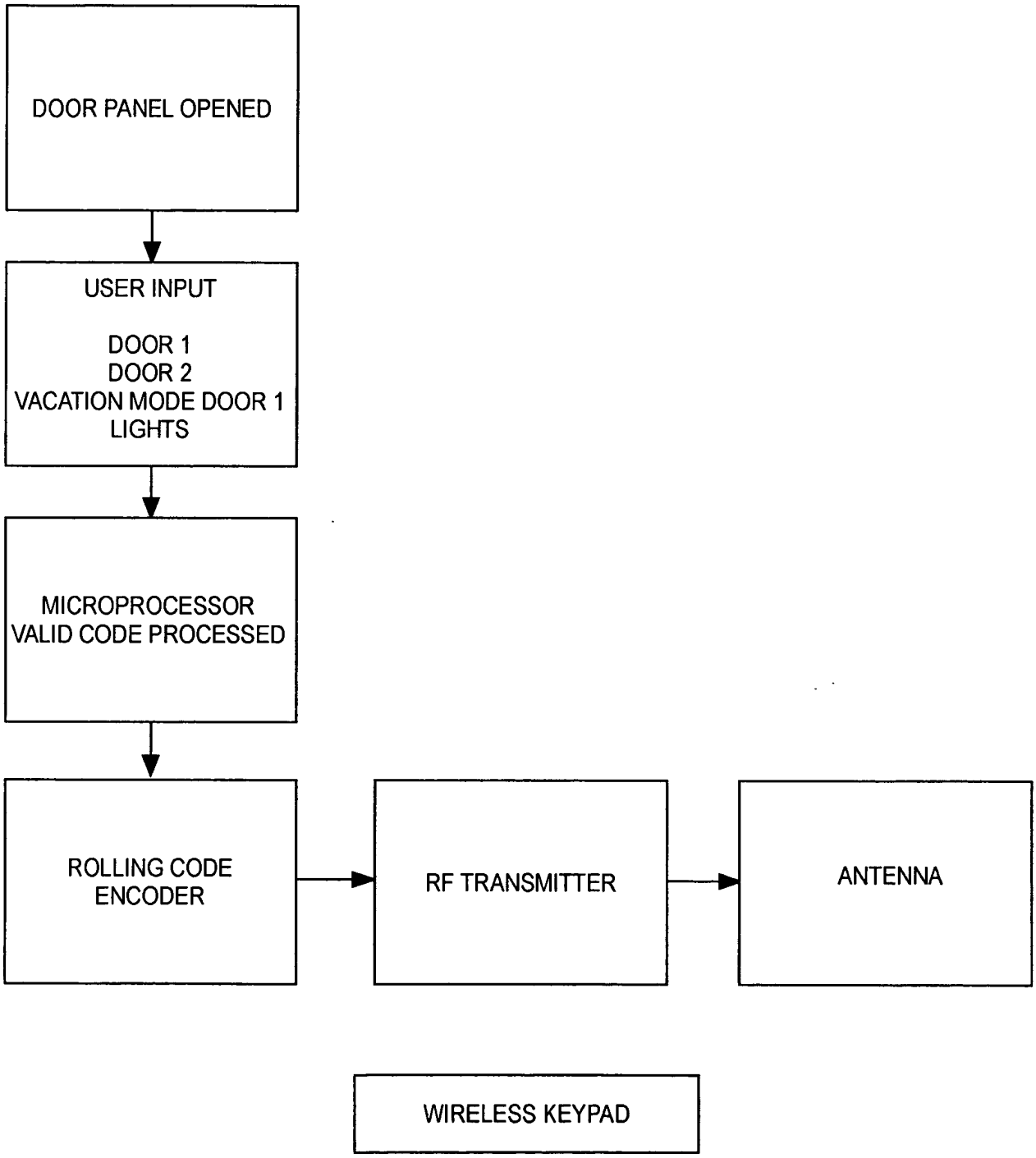

DOOR PANEL OPENED
USER INPUT
DOOR 1
DOOR 2
VACATION MODE DOOR 1
LIGHTS
MICROPROCESSOR
VALID CODE PROCESSED
ROLLING CODE
ENCODER
RF TRANSMITTER
ANTENNA
WIRELESS KEYPAD

Fig. 39

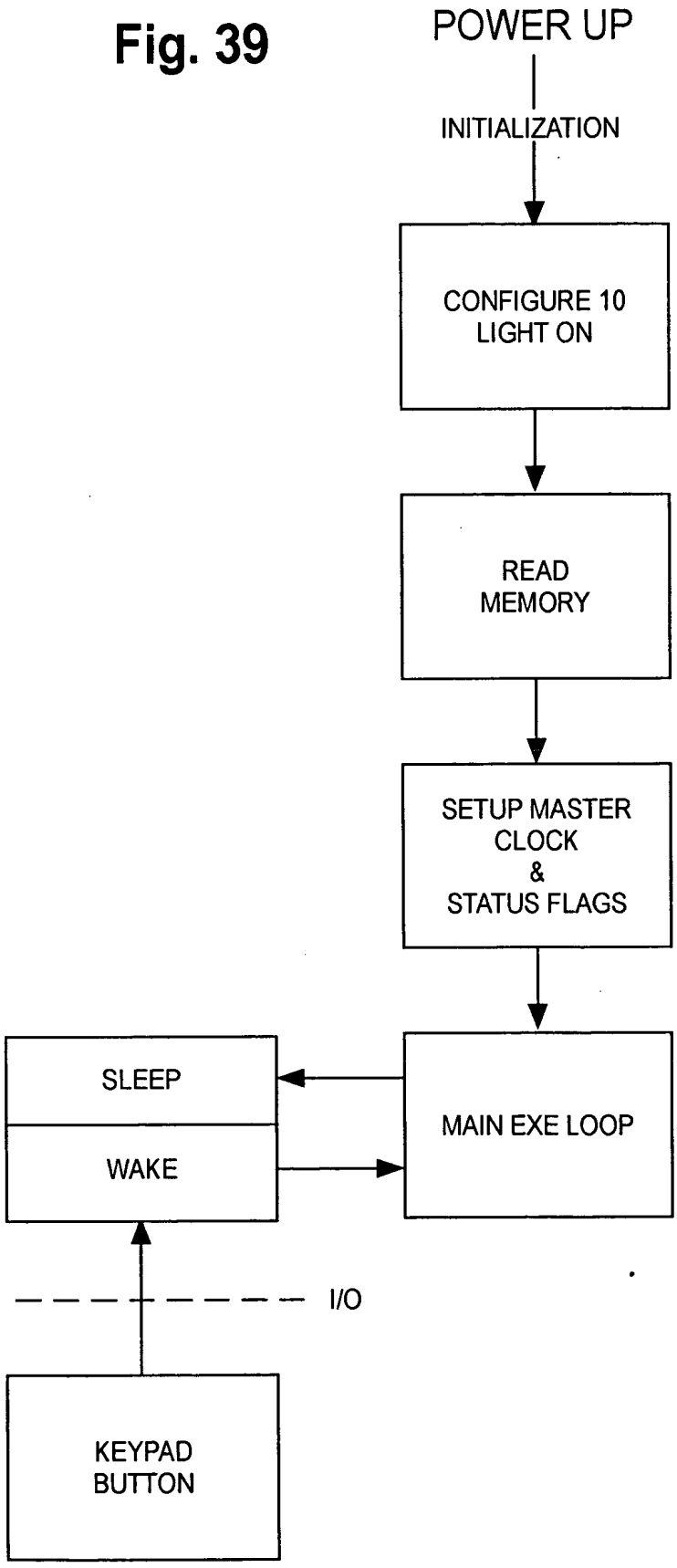

POWER UP
INITIALIZATION
CONFIGURE 10 LIGHT ON
READ MEMORY
SETUP MASTER CLOCK & STATUS FLAGS
SLEEP
WAKE
MAIN EXE LOOP
I/O
KEYPAD BUTTON

Fig. 40

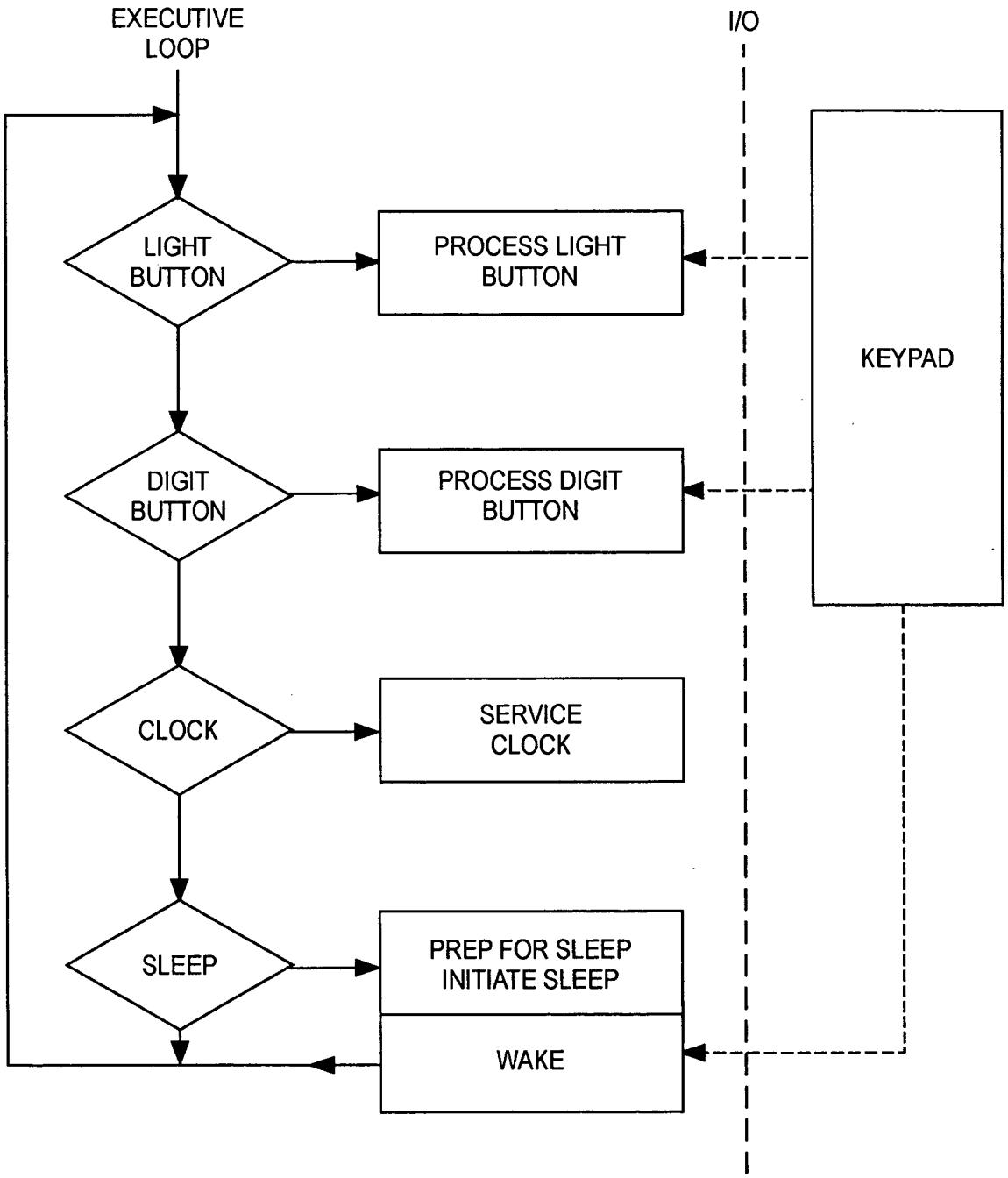

EXECUTIVE LOOP
I/O
LIGHT BUTTON
PROCESS LIGHT BUTTON
KEYPAD
DIGIT BUTTON
PROCESS DIGIT BUTTON
CLOCK
SERVICE CLOCK
SLEEP
PREP FOR SLEEP
INITIATE SLEEP
WAKE

Fig. 41

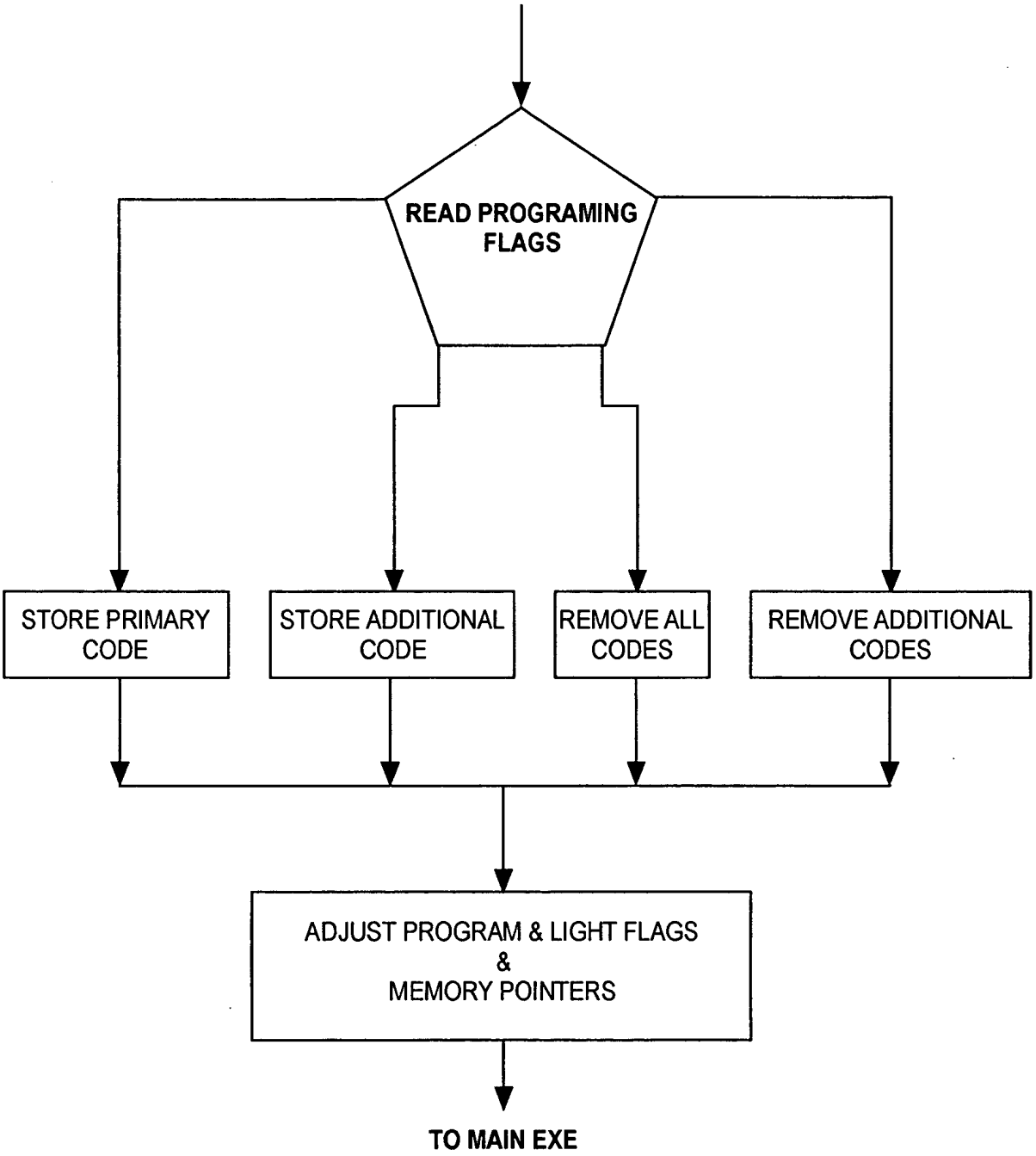

READ PROGRAMING FLAGS
STORE PRIMARY CODE
STORE ADDITIONAL CODE
REMOVE ALL CODES
REMOVE ADDITIONAL CODES
ADJUST PROGRAM & LIGHT FLAGS & MEMORY POINTERS
TO MAIN EXE

Fig. 42

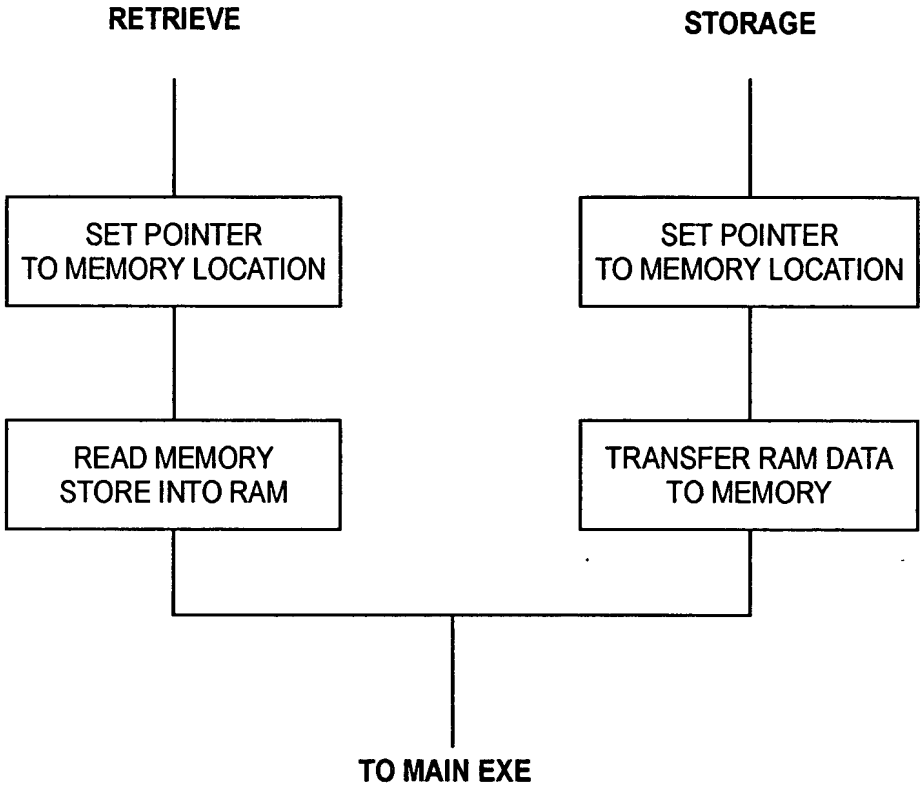

RETRIEVE
STORAGE
SET POINTER TO MEMORY LOCATION
SET POINTER TO MEMORY LOCATION
READ MEMORY STORE INTO RAM
TRANSFER RAM DATA TO MEMORY
TO MAIN EXE

Fig. 43

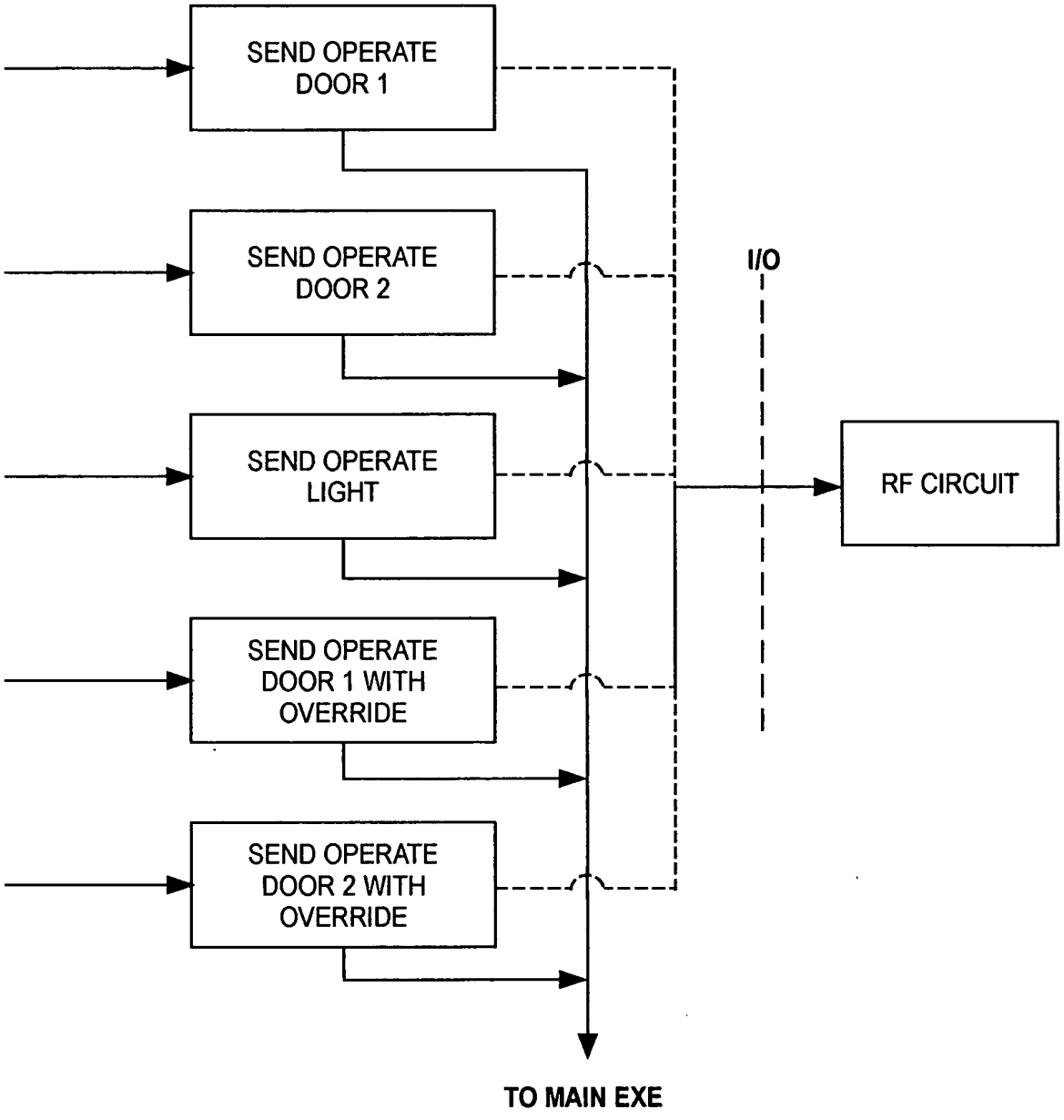

SEND OPERATE DOOR 1
SEND OPERATE DOOR 2
SEND OPERATE LIGHT
SEND OPERATE DOOR 1 WITH OVERRIDE
SEND OPERATE DOOR 2 WITH OVERRIDE
I/O
RF CIRCUIT
TO MAIN EXE

Fig. 44

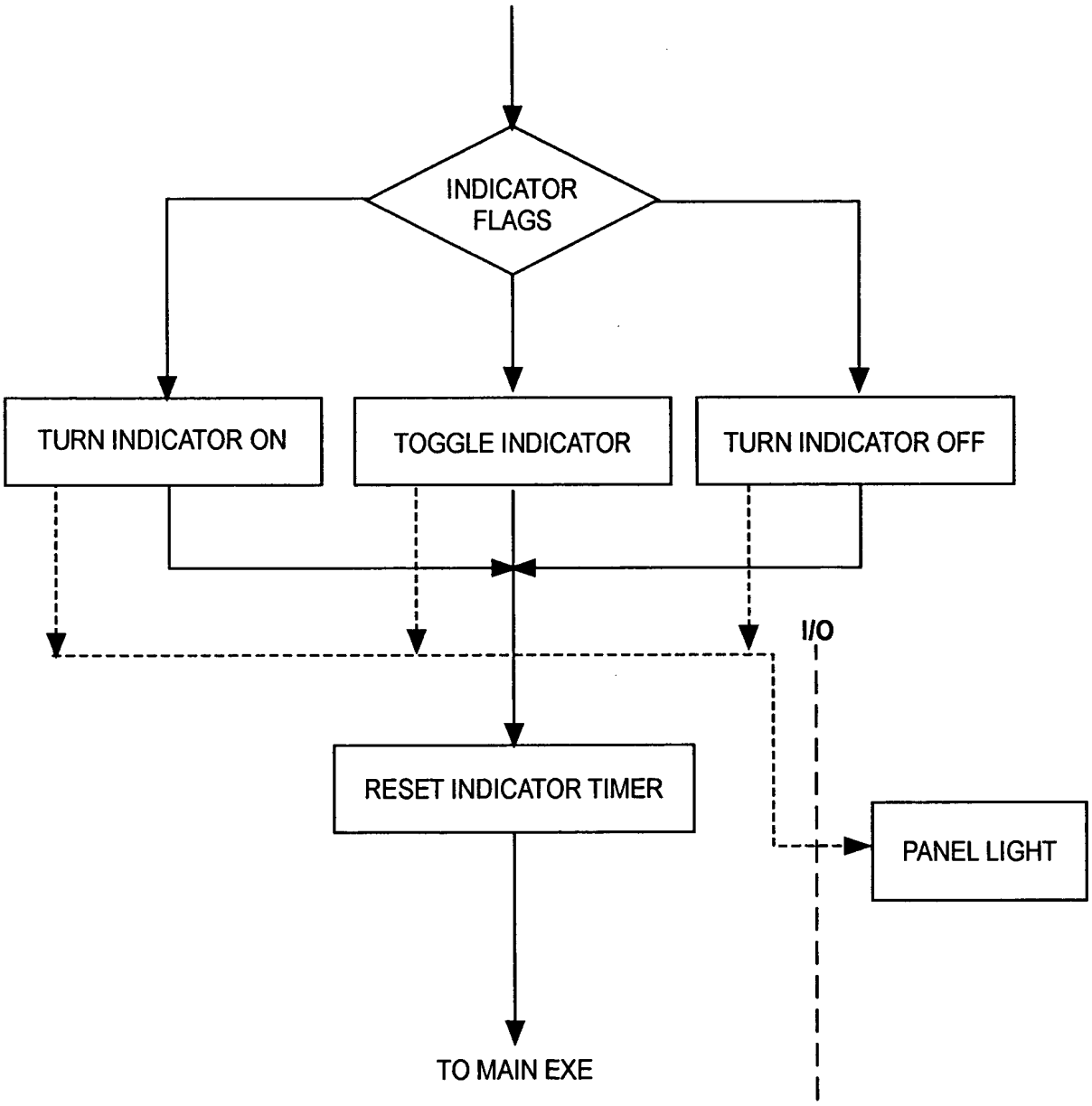
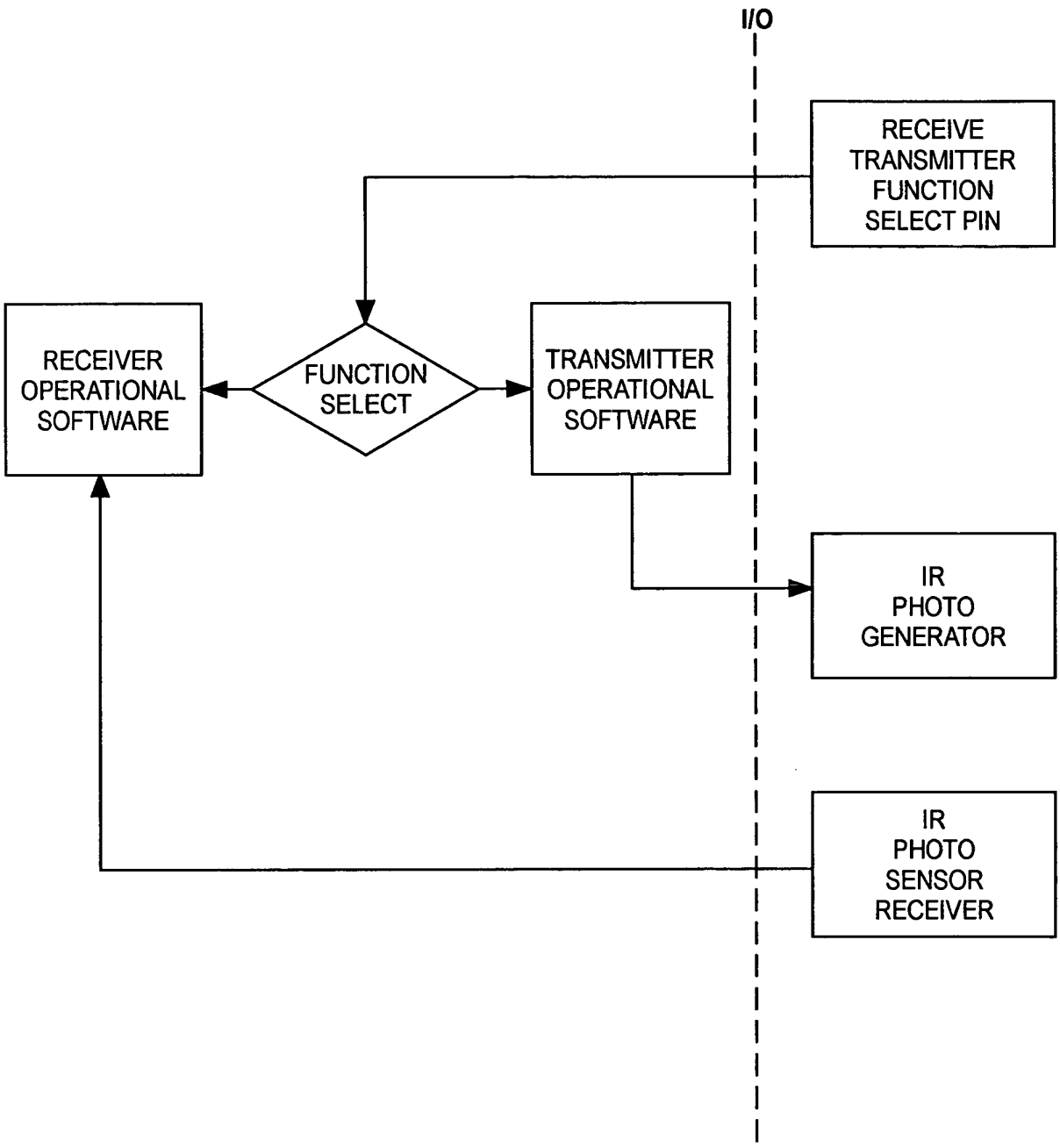

INDICATOR FLAGS
TURN INDICATOR ON
TOGGLE INDICATOR
TURN INDICATOR OFF
I/O
RESET INDICATOR TIMER
PANEL LIGHT
TO MAIN EXE

Fig. 45


I/O
RECEIVE TRANSMITTER FUNCTION SELECT PIN
RECEIVER OPERATIONAL SOFTWARE
FUNCTION SELECT
TRANSMITTER OPERATIONAL SOFTWARE
IR PHOTO GENERATOR
IR PHOTO SENSOR RECEIVER